

(19) World Intellectual Property Organization
International Bureau(43) International Publication Date
29 March 2001 (29.03.2001)

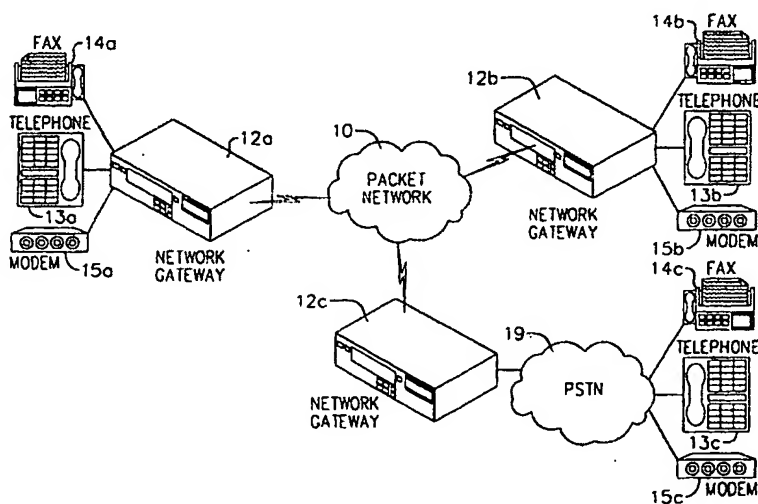
PCT

(10) International Publication Number
WO 01/22710 A2

(51) International Patent Classification ⁷ :	H04M 7/00	60/164,689	10 November 1999 (10.11.1999)	US
		60/166,289	18 November 1999 (18.11.1999)	US
(21) International Application Number:	PCT/US00/25739	09/454,219	9 December 1999 (09.12.1999)	US
		60/171,203	15 December 1999 (15.12.1999)	US
(22) International Filing Date:		60/171,180	16 December 1999 (16.12.1999)	US
	20 September 2000 (20.09.2000)	60/171,169	16 December 1999 (16.12.1999)	US
		60/171,184	16 December 1999 (16.12.1999)	US
(25) Filing Language:	English	60/178,258	25 January 2000 (25.01.2000)	US
		09/522,185	9 March 2000 (09.03.2000)	US
(26) Publication Language:	English			
(30) Priority Data:		(71) Applicant (for all designated States except US):	BROAD-COM CORPORATION [US/US]; 16215 Alton Parkway, Irvine, CA 92618 (US).	
60/154,903	20 September 1999 (20.09.1999)	US		
60/156,266	27 September 1999 (27.09.1999)	US		
60/157,470	1 October 1999 (01.10.1999)	US		
60/160,124	18 October 1999 (18.10.1999)	US		
60/161,152	22 October 1999 (22.10.1999)	US		
60/162,315	28 October 1999 (28.10.1999)	US		
60/163,169	2 November 1999 (02.11.1999)	US		
60/163,170	2 November 1999 (02.11.1999)	US		
60/163,600	4 November 1999 (04.11.1999)	US		
60/164,379	9 November 1999 (09.11.1999)	US		
60/164,690	10 November 1999 (10.11.1999)	US		
		(72) Inventors; and		
		(75) Inventors/Applicants (for US only):	LI, Henry [CA/CA];	
			200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). ENNS, David, M. [CA/CA];	
			200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). NICOL, Jordan, J. [CA/CA];	
			200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). KWAN, Kenny, C. [CA/CA];	
			200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). MITCHELL, Ross [CA/CA];	

[Continued on next page]

(54) Title: VOICE AND DATA EXCHANGE OVER A PACKET BASED NETWORK



(57) Abstract: A signal processing system which discriminates between voice signals and data signals modulated by a voiceband carrier. The signal processing system includes a voice exchange, a data exchange and a call discriminator. The voice exchange is capable of exchanging voice signals between a switched circuit network and a packet based network. The signal processing system also includes a data exchange capable of exchanging data signals modulated by a voiceband carrier on the switched circuit network with unmodulated data signal packets on the packet based network. The data exchange is performed by demodulating data signals from the switched circuit network for transmission on the packet based network, and modulating data signal packets from the packet based network for transmission on the switched circuit network. The call discriminator is used to selectively enable the voice exchange and data exchange.



200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). **LEBLANC, Wilf** [CA/CA]; 200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). **UNGER, Ken** [CA/CA]; 200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). **PAYTON, John** [CA/CA]; 200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). **STEVENSON, Shawn** [CA/CA]; 200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA). **BOORA, Bill** [CA/CA]; 200-13711 International Place, Richmond V6V 2Z8, British Columbia (CA). **TACKIN, Onur** [CA/CA]; 200-13711 International Place, Richmond, British Columbia V6V 2Z8 (CA).

(74) Agent: **GELFOUND, Craig, A.**; Christie, Parker & Hale LLP, 350 West Colorado Boulevard, P.O. Box 7068, Pasadena, CA 91109-7068 (US).

(81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ,

DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— Without international search report and to be republished upon receipt of that report.

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

1 VOICE AND DATA EXCHANGE OVER A PACKET BASED NETWORK

FIELD OF THE INVENTION

5 The present invention relates generally to telecommunications systems, and more particularly, to a system for interfacing telephony devices with packet based networks.

BACKGROUND OF THE INVENTION

10 Telephony devices, such as telephones, analog fax machines, and data modems, have traditionally utilized circuit switched networks to communicate. With the current state of technology, it is desirable for telephony devices to communicate over the Internet, or other packet based networks. Heretofore, an integrated system for interfacing various telephony devices over packet based networks has been difficult due to the different modulation schemes of the telephony devices. Accordingly, it would be advantageous to have an efficient and robust integrated system for the exchange of voice, fax data and modem data between telephony devices and packet based networks.

SUMMARY OF THE INVENTION

20 In one aspect of the present invention, a signal processing system includes a voice exchange capable of exchanging voice signals between a network line and a packet based network, and a full duplex data exchange capable of exchanging data signals from the network line with demodulated data signals from the packet based network.

25 In another aspect of the invention a signal processing system includes a voice exchange capable of exchanging voice signals between a first telephony device and a packet based network, a full duplex data exchange capable of exchanging data signals from a second telephony device with demodulated data signals from the packet based network, and a call discriminator which selectively enables at least one of the voice exchange and the data exchange.

30 In yet another aspect of the present invention, a method of processing signals includes exchanging voice signals between a network line and a packet based network, and simultaneously exchanging data signals from the network line with demodulated data signals from the packet based network.

35 In still yet another aspect of the present invention, a method of processing signals includes exchanging voice signals between a first telephony device and a packet based network, simultaneously exchanging data signals from a second telephony device with demodulated data signals from the packet based network, and discriminating between the voice signals and the data signals, and invoking at least one of the voice exchange and the data exchange based on said discrimination.

In still yet another aspect of the present invention, a signal transmission system includes a first telephony device which transmits and receives voice signals, a second telephony device

1 different from the first telephony device, a packet based network, and a signal processing system coupling the first and the second telephony devices to the packet based network, the signal processing system comprising a full duplex data exchange which exchanges data signals from the second telephony device with demodulated data signals from the packet based network.

5 It is understood that other embodiments of the present invention will become readily apparent to those skilled in the art from the following detailed description, wherein it is shown and described only embodiments of the invention by way of illustration of the best modes contemplated for carrying out the invention. As will be realized, the invention is capable of other and different embodiments and its several details are capable of modification in various other
10 respects, all without departing from the spirit and scope of the present invention. Accordingly, the drawings and detailed description are to be regarded as illustrative in nature and not as restrictive.

DESCRIPTION OF THE DRAWINGS

15 These and other features, aspects, and advantages of the present invention will become better understood with regard to the following description, appended claims, and accompanying drawings where:

FIG. 1 is a block diagram of packet based infrastructure providing a communication medium with a number of telephony devices in accordance with a preferred embodiment of the
20 present invention;

FIG. 2 is a block diagram of a signal processing system implemented with a programmable digital signal processor (DSP) software architecture in accordance with a preferred embodiment of the present invention;

FIG. 3 is a block diagram of the software architecture operating on the DSP platform of
25 FIG. 2 in accordance with a preferred embodiment of the present invention;

FIG. 4 is state machine diagram of the operational modes of a virtual device driver for packet based network applications in accordance with a preferred embodiment of the present invention;

FIG. 5 is a block diagram of several signal processing systems in the voice mode for
30 interfacing between a switched circuit network and a packet based network in accordance with a preferred embodiment of the present invention;

FIG. 6 is a system block diagram of a signal processing system operating in a voice mode in accordance with a preferred embodiment of the present invention;

FIG. 7 is a block diagram of a method for canceling echo returns in accordance with a
35 preferred embodiment of the present invention;

FIG. 8A is a block diagram of a method for normalizing the power level of a digital voice samples to ensure that the conversation is of an acceptable loudness in accordance with a preferred embodiment of the present invention;

1 FIG. 8B is a graphical depiction of a representative output of a peak tracker as a function
of a typical input signal, demonstrating that the reference value that the peak tracker forwards to
a gain calculator to adjust the power level of digital voice samples should preferably rise quickly
if the signal amplitude increases, but decrement slowly if the signal amplitude decreases in
5 accordance with a preferred embodiment of the present invention;

 FIG. 9 is a graphical depiction of exemplary operating thresholds for adjusting the gain
factor applied to digital voice samples to ensure that the conversation is of an acceptable loudness
in accordance with a preferred embodiment of the present invention;

10 FIG. 10 is a block diagram of a method for estimating the spectral shape of the
background noise of a voice transmission in accordance with a preferred embodiment of the
present invention;

 FIG. 11 is a block diagram of a method for generating comfort noise with an energy level
and spectral shape that substantially matches the background noise of a voice transmission in
accordance with a preferred embodiment of the present invention;

15 FIG. 12 is a block diagram of the voice decoder and the lost packet recovery engine in
accordance with a preferred embodiment of the present invention;

 FIG. 13A is a flow chart of the preferred lost frame recovery algorithm in accordance with
a preferred embodiment of the present invention;

20 FIG. 13B is a flow chart of the voicing decision and pitch period calculation in accordance
with a preferred embodiment of the present invention;

 FIG. 13C is a flow chart demonstrating voicing synthesis performed when packets are lost
and for the first decoded voice packet after a series of lost packets in accordance with a preferred
embodiment of the present invention;

25 FIG. 14 is a block diagram of a method for detecting dual tone multi frequency tones in
accordance with a preferred embodiment of the present invention;

 FIG. 14A is a block diagram of a method for reducing the instructions required to detect
a valid dual tone and for pre-detecting a dual tone;

 FIG. 15 is a block diagram of a signaling service for detecting precise tones in accordance
with a preferred embodiment of the present invention;

30 FIG. 16 is a block diagram of a method for detecting the frequency of a precise tone in
accordance with a preferred embodiment of the present invention;

 FIG. 17 is state machine diagram of a power state machine which monitors the estimated
power level within each of the precise tone frequency bands in accordance with a preferred
embodiment of the present invention;

35 FIG. 18 is state machine diagram of a cadence state machine for monitoring the cadence
(on/off times) of a precise tone in a voice signal in accordance with a preferred embodiment of
the present invention;

 FIG. 18A is a block diagram of a cadence processor for detecting precise tones in

1 accordance with a preferred embodiment of the present invention;

FIG. 19 is a block diagram of resource manager interface with several VHD's and PXD's in accordance with a preferred embodiment of the present invention;

5 FIG. 20 is a block diagram of several signal processing systems in the fax relay mode for interfacing between a switched circuit network and a packet based network in accordance with a preferred embodiment of the present invention;

FIG. 21 is a system block diagram of a signal processing system operating in a real time fax relay mode in accordance with a preferred embodiment of the present invention;

10 FIG. 22 is a diagram of the message flow for a fax relay in non error control mode in accordance with a preferred embodiment of the present invention;

FIG. 23 is a flow diagram of a method for fax mode spoofing in accordance with a preferred embodiment of the present invention;

15 FIG. 24 is a block diagram of several signal processing systems in the modem relay mode for interfacing between a switched circuit network and a packet based network in accordance with a preferred embodiment of the present invention;

FIG. 25 is a system block diagram of a signal processing system operating in a modem relay mode in accordance with a preferred embodiment of the present invention;

FIG. 26 is a diagram of a relay sequence for V.32bis rate synchronization using rate re-negotiation in accordance with a preferred embodiment of the present invention;

20 FIG. 27 is a diagram of an alternate relay sequence for V.32bis rate synchronization whereby rate signals are used to align the connection rates at the two ends of the network without rate re-negotiation in accordance with a preferred embodiment of the present invention;

FIG. 28 is a system block diagram of a QAM data pump transmitter in accordance with a preferred embodiment of the present invention;

25 FIG. 29 is a system block diagram of a QAM data pump receiver in accordance with a preferred embodiment of the present invention;

FIG. 30 is a block diagram of a method for sampling a signal of symbols received in a data pump receiver in synchronism with the transmitter clock of a data pump transmitter in accordance with a preferred embodiment of the present invention;

30 FIG. 31 is a block diagram of a second order loop filter for reducing symbol clock jitter in the timing recovery system of data pump receiver in accordance with a preferred embodiment of the present invention;

35 FIG. 32 is a block diagram of an alternate method for sampling a signal of symbols received in a data pump receiver in synchronism with the transmitter clock of a data pump transmitter in accordance with a preferred embodiment of the present invention;

FIG. 33 is a block diagram of an alternate method for sampling a signal of symbols received in a data pump receiver in synchronism with the transmitter clock of a data pump transmitter wherein a timing frequency offset compensator provides a fixed dc component to

1 compensate for clock frequency offset present in the received signal in accordance with a preferred embodiment of the present invention;

FIG. 34 is a block diagram of a method for estimating the timing frequency offset required to sample a signal of symbols received in a data pump receiver in synchronism with the transmitter clock of a data pump transmitter in accordance with a preferred embodiment of the present invention;

FIG. 35 is a block diagram of a method for adjusting the gain of a data pump receiver (fax or modem) to compensate for variations in transmission channel conditions; and

FIG. 36 is a block diagram of a method for detecting human speech in a telephony signal.

10 DETAILED DESCRIPTION OF THE INVENTION

An Embodiment of a Signal Processing System

In a preferred embodiment of the present invention, a signal processing system is employed to interface telephony devices with packet based networks. Telephony devices include, by way of example, analog and digital phones, ethernet phones, Internet Protocol phones, fax machines, data modems, cable modems, interactive voice response systems, PBXs, key systems, and any other conventional telephony devices known in the art. The described preferred embodiment of the signal processing system can be implemented with a variety of technologies including, by way of example, embedded communications software that enables transmission of information, including voice, fax and modem data over packet based networks. The embedded communications software is preferably run on programmable digital signal processors (DSPs) and is used in gateways, cable modems, remote access servers, PBXs, and other packet based network appliances.

An exemplary topology is shown in FIG. 1 with a packet based network 10 providing a communication medium between various telephony devices. Each network gateway 12a, 12b, 12c includes a signal processing system which provides an interface between the packet based network 10 and a number of telephony devices. In the described exemplary embodiment, each network gateway 12a, 12b, 12c supports a fax machine 14a, 14b, 14c, a telephone 13a, 13b, 13c, and a modem 15a, 15b, 15c. As will be appreciated by those skilled in the art, each network gateway 12a, 12b, 12c could support a variety of different telephony arrangements. By way of example, each network gateway might support any number telephony devices and/or circuit switched / packet based networks including, among others, analog telephones, ethernet phones, fax machines, data modems, PSTN lines (Public Switching Telephone Network), ISDN lines (Integrated Services Digital Network), T1 systems, PBXs, key systems, or any other conventional telephony device and/or circuit switched/ packet based network. In the described exemplary embodiment, two of the network gateways 12a, 12b provide a direct interface between their respective telephony devices and the packet based network 10. The other network gateway 12c is connected to its respective telephony device through a PSTN 19. The network gateways 12a,

1 12b, 12c permit voice, fax and modem data to be carried over packet based networks such as PCs
running through a USB (Universal Serial Bus) or an asynchronous serial interface, Local Area
Networks (LAN) such as Ethernet, Wide Area Networks (WAN) such as Internet Protocol (IP),
5 Frame Relay (FR), Asynchronous Transfer Mode (ATM), Public Digital Cellular Network such
as TDMA (IS-13x), CDMA (IS-9x) or GSM for terrestrial wireless applications, or any other
packet based system.

The exemplary signal processing system can be implemented with a programmable DSP
software architecture as shown in FIG. 2. This architecture has a DSP 17 with memory 18 at the
core, a number of network channel interfaces 19 and telephony interfaces 20, and a host 21 that
10 may reside in the DSP itself or on a separate microcontroller. The network channel interfaces
19 provide multi-channel access to the packet based network. The telephony interfaces 23 can
be connected to a circuit switched network interface such as a PSTN system, or directly to any
telephony device. The programmable DSP is effectively hidden within the embedded
communications software layer. The software layer binds all core DSP algorithms together,
15 interfaces the DSP hardware to the host, and provides low level services such as the allocation
of resources to allow higher level software programs to run.

An exemplary multi-layer software architecture operating on a DSP platform is shown in
FIG.3. A user application layer 26 provides overall executive control and system management,
and directly interfaces a DSP server 25 to the host 21 (see to FIG. 2). The DSP server 25
20 provides DSP resource management and telecommunications signal processing. Operating below
the DSP server layer are a number of physical devices (PXD) 30a, 30b, 30c. Each PXD provides
an interface between the DSP server 25 and an external telephony device (not shown) via a
hardware abstraction layer (HAL) 34.

The DSP server 25 includes a resource manager 24 which receives commands from,
25 forwards events to, and exchanges data with the user application layer 26. The user application
layer 26 can either be resident on the DSP 17 or alternatively on the host 21 (see FIG. 2), such
as a microcontroller. An application programming interface 27 (API) provides a software
interface between the user application layer 26 and the resource manager 24. The resource
manager 24 manages the internal / external program and data memory of the DSP 17. In addition
30 the resource manager dynamically allocates DSP resources, performs command routing as well
as other general purpose functions.

The DSP server 25 also includes virtual device drivers (VHDs) 22a, 22b, 22c. The VHDs
are a collection of software objects that control the operation of and provide the facility for real
time signal processing. Each VHD 22a, 22b, 22c includes an inbound and outbound media
35 queue (not shown) and a library of signal processing services specific to that VHD 22a, 22b, 22c.
In the described exemplary embodiment, each VHD 22a, 22b, 22c is a complete self-contained
software module for processing a single channel with a number of different telephony devices.
Multiple channel capability can be achieved by adding VHDs to the DSP server 25. The resource

1 manager 24 dynamically controls the creation and deletion of VHDs and services.

A switchboard 32 in the DSP server 25 dynamically inter-connects the PXDs 30a, 30b, 30c with the VHDs 22a, 22b, 22c. Each PXD 30a, 30b, 30c is a collection of software objects which provide signal conditioning for one external telephony device. For example, a PXD may
5 provide volume and gain control for signals from a telephony device prior to communication with the switchboard 32. Multiple telephony functionalities can be supported on a single channel by connecting multiple PXDs, one for each telephony device, to a single VHD via the switchboard 32. Connections within the switchboard 32 are managed by the user application layer 26 via a set of API commands to the resource manager 24. The number of PXDs and
10 VHDs is expandable, and limited only by the memory size and the MIPS (millions instructions per second) of the underlying hardware.

A hardware abstraction layer (HAL) 34 interfaces directly with the underlying DSP 17 hardware (see FIG. 2) and exchanges telephony signals between the external telephony devices and the PXDs. The HAL 34 includes basic hardware interface routines, including DSP
15 initialization, target hardware control, codec sampling, and hardware control interface routines. The DSP initialization routine is invoked by the user application layer 26 to initiate the initialization of the signal processing system. The DSP initialization sets up the internal registers of the signal processing system for memory organization, interrupt handling, timer initialization, and DSP configuration. Target hardware initialization involves the initialization of all hardware
20 devices and circuits external to the signal processing system. The HAL 34 is a physical firmware layer that isolates the communications software from the underlying hardware. This methodology allows the communications software to be ported to various hardware platforms by porting only the affected portions of the HAL 34 to the target hardware.

The exemplary software architecture described above can be integrated into numerous
25 telecommunications products. In an exemplary embodiment, the software architecture is designed to support telephony signals between telephony devices (and/or circuit switched networks) and packet based networks. A network VHD (NetVHD) is used to provide a single channel of operation and provide the signal processing services for transparently managing voice, fax, and modem data across a variety of packet based networks. More particularly, the NetVHD
30 encodes and packetizes DTMF, voice, fax, and modem data received from various telephony devices and/or circuit switched networks and transmits the packets to the user application layer. In addition, the NetVHD disassembles DTMF, voice, fax, and modem data from the user application layer, decodes the packets into signals, and transmits the signals to the circuit switched network or device.

35 An exemplary embodiment of the NetVHD operating in the described software architecture is shown in FIG. 4. The NetVHD includes four operational modes, namely voice mode 36, voiceband data mode 37, fax relay mode 40, and data relay mode 42. In each operational mode, the resource manager invokes various services. For example, in the voice

1 mode 36, the resource manager invokes call discrimination 44, packet voice exchange 48, and
packet tone exchange 50. The packet voice exchange 48 may employ numerous voice
compression algorithms, including, among others, Linear 128 kbps, G.711 u-law/A-law 64 kbps
5 (ITU Recommendation G.711 (1988) - Pulse code modulation (PCM) of voice frequencies),
G.726 16/24/32/40 kbps (ITU Recommendation G.726 (12/90) - 40, 32, 24, 16 kbit/s Adaptive
Differential Pulse Code Modulation (ADPCM)), G.729A 8 kbps (Annex A (11/96) to ITU
Recommendation G.729 - Coding of speech at 8 kbit/s using conjugate structure algebraic-code-
excited linear-prediction (CS-ACELP) - Annex A: Reduced complexity 8 kbit/s CS-ACELP
10 speech codec), and G.723 5.3/6.3 kbps (ITU Recommendation G.723.1 (03/96) - Dual rate coder
for multimedia communications transmitting at 5.3 and 6.3 kbit/s). The contents of each of the
foregoing ITU Recommendations being incorporated herein by reference as if set forth in full.

The packet voice exchange 48 is common to both the voice mode 36 and the voiceband
data mode 37. In the voiceband data mode 37, the resource manager invokes the packet voice
exchange 48 for exchanging transparently data without modification (other than packetization)
15 between the telephony device (or circuit switched network) and the packet based network. This
is typically used for the exchange of fax and modem data when bandwidth concerns are minimal
as an alternative to demodulation and remodulation. During the voiceband data mode 37, the
human speech detector service 59 is also invoked by the resource manager. The human speech
detector 59 monitors the signal from the near end telephony device for speech. In the event that
20 speech is detected by the human speech detector 59, an event is forwarded to the resource
manager which, in turn, causes the resource manager to terminate the human speech detector
service 59 and invoke the appropriate services for the voice mode 36 (i.e., the call discriminator,
the packet tone exchange, and the packet voice exchange).

In the fax relay mode 40, the resource manager invokes a fax exchange 52 service. The
25 packet fax exchange 52 may employ various data pumps including, among others, V.17 which
can operate up to 14,400 bits per second, V.29 which uses a 1700-Hz carrier that is varied in both
phase and amplitude, resulting in 16 combinations of 8 phases and 4 amplitudes which can
operate up to 9600 bits per second, and V.27ter which can operate up to 4800 bits per second.
Likewise, the resource manager invokes a packet data exchange 54 service in the data relay mode
30 42. The packet data exchange 52 may employ various data pumps including, among others,
V.22bis/V.22 with data rates up to 2400 bits per second, V.32bis/V.32 which enables full-duplex
transmission at 14,400 bits per second, and V.34 which operates up to 33,600 bits per second.
The ITU Recommendations setting forth the standards for the foregoing data pumps are
incorporated herein by reference as if set forth in full.

35 In the described exemplary embodiment, the user application layer does not need to
manage any service directly. The user application layer manages the session using high-level
commands directed to the NetVHD, which in turn directly runs the services. However, the user

1 application layer can access more detailed parameters of any service if necessary to change, by way of example, default functions for any particular application.

5 In operation, the user application layer opens the NetVHD and connects it to the appropriate PXD. The user application then may configure various operational parameters of the NetVHD, including, among others, default voice compression (Linear, G.711, G.726, G.723.1, G.723.1A, G.729A, G.729B), fax data pump (Binary, V.17, V.29, V.27ter), and modem data pump (Binary, V.22bis, V.32bis, V.34). The user application layer then loads an appropriate signaling service (not shown) into the NetVHD, configures it and sets the NetVHD to the On-hook state.

10 In response to events from the signaling service (not shown) via a near end telephony device (hookswitch), or signal packets from the far end, the user application will set the NetVHD to the appropriate off-hook state, typically voice mode. In an exemplary embodiment, if the signaling service event is triggered by the near end telephony device, the packet tone exchange will generate dial tone. Once a DTMF tone is detected, the dial tone is terminated. The DTMF tones are packetized and forwarded to the user application layer for transmission on the packet based network. The packet tone exchange could also play ringing tone back to the near end telephony device (when a far end telephony device is being rung), and a busy tone if the far end telephony device is unavailable. Other tones may also be supported to indicate all circuits are busy, or an invalid sequence of DTMF digits were entered on the near end telephony device.

20 Once a connection is made between the near end and far end telephony devices, the call discriminator is responsible for differentiating between a voice and machine call by detecting the presence of a 2100 Hz. tone (as in the case when the telephony device is a fax or a modem), a 1100 Hz. tone or V.21 modulated high level data link control (HDLC) flags (as in the case when the telephony device is a fax). If a 1100 Hz. tone, or V.21 modulated HDLC flags are detected, a calling fax machine is recognized. The NetVHD then terminates the voice mode 36 and invokes the packet fax exchange to process the call. If however, 2100 Hz tone is detected, the NetVHD terminates voice mode and invokes the packet data exchange.

25 The packet data exchange service further differentiates between a fax and modem by continuing to monitor the incoming signal for V.21 modulated HDLC flags, which if present, indicate that a fax connection is in progress. If HDLC flags are detected, the NetVHD terminates packet data exchange service and initiates packet fax exchange service. Otherwise, the packet data exchange service remains operative. In the absence of an 1100 or 2100 Hz. tone, or V.21 modulated HDLC flags the voice mode remains operative.

30 A. The Voice Mode

35 Voice mode provides signal processing of voice signals. As shown in the exemplary embodiment depicted in FIG. 5, voice mode enables the transmission of voice over a packet based system such as Voice over IP (VoIP, H.323), Voice over Frame Relay (VoFR, FRF-11), Voice Telephony over ATM (VTOA), or any other proprietary network. The voice mode should

1 also permit voice to be carried over traditional media such as time division multiplex (TDM)
networks and voice storage and playback systems. Network gateway 55a supports the exchange
of voice between a traditional circuit switched 58 and a packet based network 56. In addition,
network gateways 55b, 55c, 55d, 55e support the exchange of voice between the packet based
5 network 56 and a number of telephones 57a, 57b, 57c, 57d, 57e. Although the described
exemplary embodiment is shown for telephone communications across the packet based network,
it will be appreciated by those skilled in the art that other telephony/network devices could be
used in place of one or more of the telephones, such as a HPNA phone connected via a cable
modem.

10 The PXDs for the voice mode provide echo cancellation, gain, and automatic gain control.
The network VHD invokes numerous services in the voice mode including call discrimination,
packet voice exchange, and packet tone exchange. These network VHD services operate together
to provide: (1) an encoder system with DTMF detection, call progress tone detection, voice
activity detection, voice compression, and comfort noise estimation, and (2) a decoder system
15 with delay compensation, voice decoding, DTMF generation, comfort noise generation and lost
frame recovery.

The services invoked by the network VHD in the voice mode and the associated PXD is
shown schematically in FIG. 6. In the described exemplary embodiment, the PXD 60 provides
two way communication with a telephone or a circuit switched network, such as a PSTN line
20 (e.g. DS0) carrying a 64kb/s pulse code modulated (PCM) signal, i.e., digital voice samples.

The incoming PCM signal 60a is initially processed by the PXD 60 to remove far end
echos that might otherwise be transmitted back to the far end user. As the name implies, echos
in telephone systems is the return of the talker's voice resulting from the operation of the hybrid
with its two-four wire conversion. If there is low end-to-end delay, echo from the far end is
equivalent to side-tone (echo from the near-end), and therefore, not a problem. Side-tone gives
25 users feedback as to how loud they are talking, and indeed, without side-tone, users tend to talk
too loud. However, far end echo delays of more than about 10 to 30 msec significantly degrade
the voice quality and are a major annoyance to the user.

An echo canceller 70 is used to remove echos from far end speech present on the
30 incoming PCM signal 60a before routing the incoming PCM signal 60a back to the far end user.
The echo canceller 70 samples an outgoing PCM signal 60b from the far end user, filters it, and
combines it with the incoming PCM signal 60a. Preferably, the echo canceller 70 is followed
by a non-linear processor (NLP) 72 which may mute the digital voice samples when far end
speech is detected in the absence of near end speech. The echo canceller 70 may also inject
35 comfort noise which in the absence of near end speech may be roughly at the same level as the
true background noise or at a fixed level.

After echo cancellation, the power level of the digital voice samples is normalized by an
automatic gain control (AGC) 74 to ensure that the conversation is of an acceptable loudness.

1 Alternatively, the AGC can be performed before the echo canceller 70, however, this approach
would entail a more complex design because the gain would also have to be applied to the
sampled outgoing PCM signal 60b. In the described exemplary embodiment, the AGC 74 is
designed to adapt slowly, although it should adapt fairly quickly if overflow or clipping is
5 detected. The AGC adaptation should be held fixed if the NLP 72 is activated.

After AGC, the digital voice samples are placed in the media queue 66 in the network
VHD 62 via the switchboard 32'. In the voice mode, the network VHD 62 invokes three services,
namely call discrimination, packet voice exchange, and packet tone exchange. The call
discriminator 68 analyzes the digital voice samples from the media queue to determine whether
10 a 2100 Hz, a 1100 Hz. tone or V.21 modulated HDLC flags are present. As described above with
reference to FIG. 4, if either tone or HDLC flags are detected, the voice mode services are
terminated and the appropriate service for fax or modem operation is initiated. In the absence
of a 2100 Hz, a 1100 Hz. tone, or HDLC flags, the digital voice samples are coupled to the
encoder system which includes a voice encoder 82, a voice activity detector (VAD) 80, a comfort
15 noise estimator 81, a DTMF detector 76, a call progress tone detector 77 and a packetization
engine 78.

Typical telephone conversations have as much as sixty percent silence or inactive content.
Therefore, high bandwidth gains can be realized if digital voice samples are suppressed during
these periods. A VAD 80, operating under the packet voice exchange, is used to accomplish this
20 function. The VAD 80 attempts to detect digital voice samples that do not contain active speech.
During periods of inactive speech, the comfort noise estimator 81 couples silence identifier (SID)
packets to a packetization engine 78. The SID packets contain voice parameters that allow the
reconstruction of the background noise at the far end.

From a system point of view, the VAD 80 may be sensitive to the change in the NLP 72.
25 For example, when the NLP 72 is activated, the VAD 80 may immediately declare that voice is
inactive. In that instance, the VAD 80 may have problems tracking the true background noise
level. If the echo canceller 70 generates comfort noise during periods of inactive speech, it may
have a different spectral characteristic from the true background noise. The VAD 80 may detect
a change in noise character when the NLP 72 is activated (or deactivated) and declare the comfort
30 noise as active speech. For these reasons, the VAD 80 should be disabled when the NLP 72 is
activated. This is accomplished by a "NLP on" message 72a passed from the NLP 72 to the
VAD 80.

The voice encoder 82, operating under the packet voice exchange, can be a straight 16
bit PCM encoder or any voice encoder which supports one or more of the standards promulgated
35 by ITU. The encoded digital voice samples are formatted into a voice packet (or packets) by the
packetization engine 78. These voice packets are formatted according to an applications protocol
and outputted to the host (not shown). The voice encoder 82 is invoked only when digital voice
samples with speech are detected by the VAD 80. Since the packetization interval may be a

1 multiple of an encoding interval, both the VAD 80 and the packetization engine 78 should
cooperate to decide whether or not the voice encoder 82 is invoked. For example, if the
packetization interval is 10 msec and the encoder interval is 5 msec (a frame of digital voice
5 samples is 5 ms), then a frame containing active speech should cause the subsequent frame to be
placed in the 10 ms packet regardless of the VAD state during that subsequent frame. This
interaction can be accomplished by the VAD 80 passing an "active" flag 80a to the packetization
engine 78, and the packetization engine 78 controlling whether or not the voice encoder 82 is
invoked.

10 In the described exemplary embodiment, the VAD 80 is applied after the AGC 74. This
approach provides optimal flexibility because both the VAD 80 and the voice encoder 82 are
integrated into some speech compression schemes such as those promulgated in ITU
Recommendations G.729 with Annex B VAD (March 1996) - Coding of Speech at 8 kbits/s
Using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP), and G.723.1
with Annex A VAD (March 1996) - Dual Rate Coder for Multimedia Communications
15 Transmitting at 5.3 and 6.3 kbit/s, the contents of which is hereby incorporated by reference as
through set forth in full herein.

Operating under the packet tone exchange, a DTMF detector 76 determines whether or
not there is a DTMF signal present at the near end. The DTMF detector 76 also provides a pre-
detection flag 76a which indicates whether or not it is likely that the digital voice sample might
20 be a portion of a DTMF signal. If so, the pre-detection flag 76a is relayed to the packetization
engine 78 instructing it to begin holding voice packets. If the DTMF detector 76 ultimately
detects a DTMF signal, the voice packets are discarded, and the DTMF signal is coupled to the
packetization engine 78. Otherwise the voice packets are ultimately released from the
packetization engine 78 to the host (not shown). The benefit of this method is that there is only
25 a temporary impact on voice packet delay when a DTMF signal is pre-detected in error, and not
a constant buffering delay. Whether voice packets are held while the pre-detection flag 76a is
active could be adaptively controlled by the user application layer.

Similarly, a call progress tone detector 77 also operates under the packet tone exchange
to determine whether a precise signaling tone is present at the near end. Call progress tones are
30 those which indicate what is happening to dialed phone calls. Conditions like busy line, ringing
called party, bad number, and others each have distinctive tone frequencies and cadences
assigned them. The call progress tone detector 77 monitors the call progress state, and forwards
a call progress tone signal to the packetization engine to be packetized and transmitted across the
packet based network. The call progress tone detector may also provide information regarding
35 the near end hook status which is relevant to the signal processing tasks. If the hook status is on
hook, the VAD should preferably mark all frames as inactive, DTMF detection should be
disabled, and SID packets should only be transferred if they are required to keep the connection
alive.

1 The decoding system of the network VHD 62 essentially performs the inverse operation
of the encoding system. The decoding system of the network VHD 62 comprises a depacketizing
engine 84, a voice queue 86, a DTMF queue 88, a precision tone queue 87, a voice synchronizer
90, a DTMF synchronizer 102, a precision tone synchronizer 103, a voice decoder 96, a VAD
5 98, a comfort noise estimator 100, a comfort noise generator 92, a lost packet recovery engine
94, a tone generator 104, and a precision tone generator 105.

 The depacketizing engine 84 identifies the type of packets received from the host (i.e.,
voice packet, DTMF packet, call progress tone packet, SID packet), transforms them into frames
which are protocol independent. The depacketizing engine 84 then transfers the voice frames (or
10 voice parameters in the case of SID packets) into the voice queue 86, transfers the DTMF frames
into the DTMF queue 88 and transfers the call progress tones into the call progress tone queue
87. In this manner, the remaining tasks are, by and large, protocol independent.

 A jitter buffer is utilized to compensate for network impairments such as delay jitter
caused by packets not arriving at the same time or in the same order in which they were
15 transmitted. In addition, the jitter buffer compensates for lost packets that occur on occasion
when the network is heavily congested. In the described exemplary embodiment, the jitter buffer
for voice includes a voice synchronizer 90 that operates in conjunction with a voice queue 86 to
provide an isochronous stream of voice frames to the voice decoder 96.

 Sequence numbers embedded into the voice packets at the far end can be used to detect
20 lost packets, packets arriving out of order, and short silence periods. The voice synchronizer 90
can analyze the sequence numbers, enabling the comfort noise generator 92 during short silence
periods and performing voice frame repeats via the lost packet recovery engine 94 when voice
packets are lost. SID packets can also be used as an indicator of silent periods causing the voice
synchronizer 90 to enable the comfort noise generator 92. Otherwise, during far end active
25 speech, the voice synchronizer 90 couples voice frames from the voice queue 86 in an
isochronous stream to the voice decoder 96. The voice decoder 96 decodes the voice frames into
digital voice samples suitable for transmission on a circuit switched network, such as a 64kb/s
PCM signal for a PSTN line. The output of the voice decoder 96 (or the comfort noise generator
92 or lost packet recovery engine 94 if enabled) is written into a media queue 106 for
30 transmission to the PXD 60.

 The comfort noise generator 92 provides background noise to the near end user during
silent periods. If the protocol supports SID packets, (and these are supported for VTOA, FRF-11,
and VoIP), the comfort noise estimator at the far end encoding system should transmit SID
packets. Then, the background noise can be reconstructed by the near end comfort noise
35 generator 92 from the voice parameters in the SID packets buffered in the voice queue 86.
However, for some protocols, namely, FRF-11, the SID packets are optional, and other far end
users may not support SID packets at all. In these systems, the voice synchronizer 90 must
continue to operate properly. In the absence of SID packets, the voice parameters of the

1 background noise at the far end can be determined by running the VAD 98 at the voice decoder 96 in series with a comfort noise estimator 100.

Preferably, the voice synchronizer 90 is not dependent upon sequence numbers embedded in the voice packet. The voice synchronizer 90 can invoke a number of mechanisms to
5 compensate for delay jitter in these systems. For example, the voice synchronizer 90 can assume that the voice queue 86 is in an underflow condition due to excess jitter and perform packet repeats by enabling the lost frame recovery engine 94. Alternatively, the VAD 98 at the voice decoder 96 can be used to estimate whether or not the underflow of the voice queue 86 was due to the onset of a silence period or due to packet loss. In this instance, the spectrum and/or the
10 energy of the digital voice samples can be estimated and the result 98a fed back to the voice synchronizer 90. The voice synchronizer 90 can then invoke the lost packet recovery engine 94 during voice packet losses and the comfort noise generator 92 during silent periods.

When DTMF packets arrive, they are depacketized by the depacketizing engine 84. DTMF frames at the output of the depacketizing engine 84 are written into the DTMF queue 88.
15 The DTMF synchronizer 102 couples the DTMF frames from the DTMF queue 88 to the tone generator 104. Much like the voice synchronizer, the DTMF synchronizer 102 is employed to provide an isochronous stream of DTMF frames to the tone generator 104. Generally speaking, when DTMF packets are being transferred, voice frames should be suppressed. To some extent, this is protocol dependent. However, the capability to flush the voice queue 86 to ensure that the
20 voice frames do not interfere with DTMF generation is desirable. Essentially, old voice frames which may be queued are discarded when DTMF packets arrive. This will ensure that there is a significant gap before DTMF tones are generated. This is achieved by a "tone present" message 88a passed between the DTMF queue and the voice synchronizer 90.

The tone generator 104 converts the DTMF signals into a DTMF tone suitable for a
25 standard digital or analog telephone. The tone generator 104 overwrites the media queue 106 to prevent leakage through the voice path and to ensure that the DTMF tones are not too noisy.

There is also a possibility that DTMF tone may be fed back as an echo into the DTMF detector 76. To prevent false detection, the DTMF detector 76 can be disabled entirely (or disabled only for the digit being generated) during DTMF tone generation. This is achieved by
30 a "tone on" message 104a passed between the tone generator 104 and the DTMF detector 76. Alternatively, the NLP 72 can be activated while generating DTMF tones.

When call progress tone packets arrive, they are depacketized by the depacketizing engine 84. Call progress tone frames at the output of the depacketizing engine 84 are written into the call progress tone queue 87. The call progress tone synchronizer 103 couples the call progress
35 tone frames from the call progress tone queue 87 to a call progress tone generator 105. Much like the DTMF synchronizer, the call progress tone synchronizer 103 is employed to provide an isochronous stream of call progress tone frames to the call progress tone generator 105. And much like the DTMF tone generator, when call progress tone packets are being transferred, voice

1 frames should be suppressed. To some extent, this is protocol dependent. However, the
capability to flush the voice queue 86 to ensure that the voice frames do not interfere with call
progress tone generation is desirable. Essentially, old voice frames which may be queued are
discarded when call progress tone packets arrive to ensure that there is a significant inter-digit
5 gap before call progress tones are generated. This is achieved by a "tone present" message 87a
passed between the call progress tone queue 87 and the voice synchronizer 90.

The call progress tone generator 105 converts the call progress tone signals into a call
progress tone suitable for a standard digital or analog telephone. The call progress tone generator
105 overwrites the media queue 106 to prevent leakage through the voice path and to ensure that
10 the call progress tones are not too noisy.

The outgoing PCM signal in the media queue 106 is coupled to the PXD 60 via the
switchboard 32'. The outgoing PCM signal is coupled to an amplifier 108 before being outputted
on the PCM output line 60b.

1. Echo Canceller with NLP

15 The problem of line echos such as the reflection of the talker's voice resulting from the
operation of the hybrid with its two-four wire conversion is a common telephony problem. To
eliminate or minimize the effect of line echos in the described exemplary embodiment of the
present invention, an echo canceller with non-linear processing is used. Although echo
cancellation is described in the context of a signal processing system for packet voice exchange,
20 those skilled in the art will appreciate that the techniques described for echo cancellation are
likewise suitable for various applications requiring the cancellation of reflections, or other
undesirable signals, from a transmission line. Accordingly, the described exemplary embodiment
for echo cancellation in a signal processing system is by way of example only and not by way
of limitation.

25 In the described exemplary embodiment the echo canceller preferably complies with one
or more of the following ITU-T Recommendations G.164 (1988) - Echo Suppressors, G.165
(March 1993) - Echo Cancellers, and G.168 (April 1997)- Digital Network Echo Cancellers, the
contents of which are incorporated herein by reference as though set forth in full. The described
embodiment merges echo cancellation and echo suppression methodologies to remove the line
30 echos that are prevalent in telecommunication systems. Typically, echo cancellers are favored
over echo suppressors for superior overall performance in the presence of system noise such as,
for example, background music, double talk etc., while echo suppressors tend to perform well
over a wide range of operating conditions where clutter such as system noise is not present. The
described exemplary embodiment utilizes an echo suppressor when the energy level of the line
35 echo is below the audible threshold, otherwise an echo canceller is preferably used. The use of
an echo suppressor reduces system complexity, leading to lower overall power consumption or
higher densities (more VHDs per part or network gateway). Those skilled in the art will
appreciate that various signal characteristics such as energy, average magnitude, echo

1 characteristics, as well as information explicitly received in voice or SID packets may be used to determine when to bypass echo cancellation. Accordingly, the described exemplary embodiment for bypassing echo cancellation in a signal processing system as a function of estimated echo power is by way of example only and not by way of limitation.

5 Figure 7 shows the block diagram of an echo canceller in accordance with a preferred embodiment of the present invention. If required to support voice transmission via a T1 or other similar transmission media, a compressor 120 may compress the output 120(a) of the voice decoder system into a format suitable for the channel at R_{out} 120(b). Typically the compressor 120 provides μ -law or A-law compression in accordance with ITU-T standard G.711, although
10 linear compression or compression in accordance with alternate companding laws may also be supported. The compressed signal at R_{out} (signal that eventually makes it way to a near end ear piece/telephone receiver), may be reflected back as an input signal to the voice encoder system. An input signal 122(a) may also be in the compressed domain (if compressed by compressor 120) and, if so, an expander 122 may be required to invert the companding law to obtain a near end
15 signal 122(b). A power estimator 124 estimates a short term average power 124(a), a long term average power 124(b), and a maximum power level 124(c) for the near end signal 122(b).

An expander 126 inverts the companding law used to compress the voice decoder output signal 120(b) to obtain a reference signal 126(a). One of skill in the art will appreciate that the voice decoder output signal could alternatively be compressed downstream of the echo canceller
20 so that the expander 126 would not be required. However, to ensure that all non-linearities in the echo path are accounted for in the reference signal 126(a) it is preferable to compress / expand the voice decoder output signal 120(b). A power estimator 128 estimates a short term average power 128(a), a long term average power 128(b), a maximum power level 128(c) and a background power level 128(d) for the reference signal 126(a). The reference signal 126(a) is
25 input into a finite impulse response (FIR) filter 130. The FIR filter 130 models the transfer characteristics of a dialed telephone line circuit so that the unwanted echo may preferably be canceled by subtracting filtered reference signal 130(a) from the near end signal 122(b) in a difference operator 132.

However, for a variety of reasons, such as for example, non-linearities in the hybrid and
30 tail circuit, estimation errors, noise in the system, etc., the adaptive FIR filter 130 may not identically model the transfer characteristics of the telephone line circuit so that the echo canceller may be unable to cancel all of the resulting echo. Therefore, a non linear processor (NLP) 140 is used to suppress the residual echo during periods of far end active speech with no near end speech. During periods of inactive speech, a power estimator 138 estimates the
35 performance of the echo canceller by estimating a short term average power 138(a), a long term average power 138(b) and background power level 138(c) for an error signal 132(b) which is an output of the difference operator 132. The estimated performance of the echo canceller is one measure utilized by adaptation logic 136 to selectively enable a filter adapter 134 which controls

1 the convergence of the adaptive FIR filter 130. The adaptation logic 136 processes the estimated power levels of the reference signal (128a, 128b, 128c and 128d) the near end signal (124a, 124b and 124c) and the error signal (138a, 138b and 138c) to control the invocation of the filter adapter 134 as well as the step size to be used during adaptation.

5 In the described preferred embodiment, the echo suppressor is a simple bypass 144(a) that is selectively enabled by toggling the bypass cancellation switch 144. A bypass estimator 142 toggles the bypass cancellation switch 144 based upon the maximum power level 128(c) of the reference signal 126(a), the long term average power 138(b) of the error signal 132(b) and the long term average power 124(b) of the near end signal 122(b). One skilled in the art will appreciate that a NLP or other suppressor could be included in the bypass path 144(a), so that the described echo suppressor is by way of example only and not by way of limitation.

10 In an exemplary embodiment, the adaptive filter 130 models the transfer characteristics of the hybrid and the tail circuit of the telephone circuit. The tail length supported should preferably be at least 16 msec. The adaptive filter 130 may be a linear transversal filter or other suitable finite impulse response filter. In the described exemplary embodiment, the echo canceller preferably converges or adapts only in the absence of near end speech. Therefore, near end speech and/or noise present on the input signal 122(a) may cause the filter adapter 134 to diverge. To avoid divergence the filter adapter 134 is preferably selectively enabled by the adaptation logic 136. In addition, the time required for an adaptive filter to converge increases significantly with the number of coefficients to be determined. Reasonable modeling of the hybrid and tail circuits with a finite impulse response filter requires a large number of coefficients so that filter adaptation is typically computationally intense. In the described exemplary embodiment the DSP resources required for filter adaptation are minimized by adjusting the adaptation speed of the FIR filter 130.

20 The filter adapter 134 is preferably based upon a normalized least mean square algorithm (NLMS) as described in S. Haykin, Adaptive Filter Theory, and T. Parsons, Voice and Speech Processing, the contents of which are incorporated herein by reference as if set forth in full. The error signal 132(b) at the output of the difference operator 132 for the adaptation logic may preferably be characterized as follows:

$$30 \quad e(n) = s(n) - \sum c(j)r(n-j)$$

where $e(n)$ is the error signal at time n , $r(n)$ is the reference signal 126(a) at time n and $s(n)$ is the near end signal 122(b) at time n , and $c(j)$ are the coefficients of the transversal filter where the dimension of the transversal filter is preferably the worst case echo path length (i.e. the length of the tail circuit L) and $c(j)$, for $j=0$ to $L-1$, is given by:

$$35 \quad c(j) = c(j) + \mu * e(n) * r(n-j)$$

wherein $c(j)$ is preferably initialized to a reasonable value such as for example zero.

1 Assuming a block size of one msec (or 8 samples at a sampling rate of 8 kHz), the short term average power of the reference signal P_{ref} is the sum of the last L reference samples and the energy for the current eight samples so that

$$\mu = \frac{P_{ref}(n)}{\alpha}$$

5 where α is the adaptation step size. One of skill in the art will appreciate that the filter adaptation logic may be implemented in a variety of ways, including fixed point rather than the described floating point realization. Accordingly, the described exemplary adaptation logic is by way of example only and not by way of limitation.

To support filter adaptation the described exemplary embodiment includes the power estimator 128 that estimates the short term average power 128(a) of the reference signal 126(a) (P_{ref}). In the described exemplary embodiment the short term average power is preferably estimated over the worst case length of the echo path plus eight samples, (i.e. the length of the FIR filter L + 8 samples). In addition, the power estimator 128 computes the maximum power level 128(c) of the reference signal 126(a) (P_{refmax}) over a period of time that is preferably equal to the tail length L of the echo path. For example, putting a time index on the short term average power, so that $P_{ref}(n)$ is the power of the reference signal at time n. P_{refmax} is then characterized as:

$$P_{refmax}(n) = \max P_{ref}(j) \text{ for } j = n - L_{msec} \text{ to } j = n$$

where L_{msec} is the length of the tail in msec so that P_{refmax} is the maximum power in the reference signal P_{ref} over a length of time equal to the tail length.

The second power estimator 124 estimates the short term average power of the near end signal 122(b) (P_{near}) in a similar manner. The short term average power 138(a) of the error signal 132(b) (the output of difference operator 132), P_{err} is also estimated in a similar manner by the third power estimator 138.

25 In addition, the echo return loss (ERL), defined as the loss from R_{out} 120(b) to S_{in} 122(a) in the absence of near end speech, is periodically estimated and updated. In the described exemplary embodiment the ERL is estimated and updated about every 5-20 msec. The power estimator 128 estimates the long term average power 128(b) (P_{refERL}) of the reference signal 126(a) in the absence of near end speech. The second power estimator 124 estimates the long term average power 124(b) ($P_{nearERL}$) of the near end signal 122(b) in the absence of near end speech. The adaptation logic 136 computes the ERL by dividing the long term average power of the reference signal (P_{refERL}) by the long term average power of the near end signal ($P_{nearERL}$). The adaptation logic 136 preferably only updates the long term averages used to compute the estimated ERL if the estimated short term power level 128(a) (P_{ref}) of the reference signal 126(a) is greater than a predetermined threshold, preferably in the range of about -30 to -35 dBm0; and the estimated short term power level 128(a) (P_{ref}) of the reference signal 126(a) is preferably larger than about at least the short term average power 124(a) (P_{near}) of the near end signal 122(b) ($P_{ref} > P_{near}$ in the preferred embodiment).

1 In the preferred embodiment, the long term averages (P_{refERL} and $P_{nearERL}$) are based on a first order infinite impulse response (IIR) recursive filter, wherein the inputs to the two first order filters are P_{ref} and P_{near} .

$$P_{nearERL} = (1-\beta) * P_{nearERL} + P_{near} * \beta; \text{ and}$$

$$P_{refERL} = (1-\beta) * P_{refERL} + P_{ref} * \beta$$

where filter coefficient $\beta = 1/64$

Similarly, the adaptation logic 136 of the described exemplary embodiment characterizes the effectiveness of the echo canceller by estimating the echo return loss enhancement (ERLE). The ERLE is an estimation of the reduction in power of the near end signal 122(b) due to echo cancellation when there is no near end speech present. The ERLE is the average loss from the input 132(a) of the difference operator 132 to the output 132(b) of the difference operator 132. The adaptation logic 136 in the described exemplary embodiment periodically estimates and updates the ERLE, preferably in the range of about 5 to 20 msec. In operation, the power estimator 124 estimates the long term average power 124(b) $P_{nearERLE}$ of the near end signal 122(b) in the absence of near end speech. The power estimator 138 estimates the long term average power 138(b) $P_{errERLE}$ of the error signal 132(b) in the absence of near end speech. The adaptation logic 136 computes the ERLE by dividing the long term average power 124(a) $P_{nearERLE}$ of the near end signal 122(b) by the long term average power 138(b) $P_{errERLE}$ of the error signal 132(b). The adaptation logic 136 preferably updates the long term averages used to compute the estimated ERLE only when the estimated short term average power 128(a) (P_{ref}) of the reference signal 126(a) is greater than a predetermined threshold preferably in the range of about -30 to -35 dBm0; and the estimated short term average power 124(a) (P_{near}) of the near end signal 122(b) is large as compared to the estimated short term average power 138(a) (P_{err}) of the error signal (preferably when P_{near} is approximately greater than or equal to four times the short term average power of the error signal ($4P_{err}$)). Therefore, an ERLE of approximately 6 dB is preferably required before the ERLE tracker will begin to function.

In the preferred embodiment, the long term averages ($P_{nearERLE}$ and $P_{errERLE}$) may be based on a first order IIR (infinite impulse response) recursive filter, wherein the inputs to the two first order filters are P_{near} and P_{err} .

$$P_{nearERLE} = (1-\beta) * P_{nearERL} + P_{near} * \beta; \text{ and}$$

$$P_{errERLE} = (1-\beta) * P_{errERL} + P_{err} * \beta$$

where filter coefficient $\beta = 1/64$

It should be noted that $P_{nearERL} \neq P_{nearERLE}$ because the conditions under which each is updated are different.

To assist in the determination of whether to invoke the echo canceller and if so with what step size, the described exemplary embodiment estimates the power level of the background noise. The power estimator 128 tracks the long term energy level of the background noise 128(d) (B_{ref}) of the reference signal 126(a). The power estimator 128 utilizes a much faster time

1 constant when the input energy is lower than the background noise estimate (current output).
 With a fast time constant the power estimator 128 tends to track the minimum energy level of
 the reference signal 126(a). By definition, this minimum energy level is the energy level of the
 background noise of the reference signal B_{ref} . The energy level of the background noise of the
 5 error signal B_{err} is calculated in a similar manner. The estimated energy level of the background
 noise of the error signal (B_{err}) is not updated when the energy level of the reference signal is
 larger than a predetermined threshold (preferably in the range of about 30-35 dBm0).

In addition, the invocation of the echo canceller depends on whether near end speech is
 active. Preferably, the adaptation logic 136 declares near end speech active when three
 10 conditions are met. First, the short term average power of the error signal should preferably
 exceed a minimum threshold, preferably on the order of about -36 dBm0 ($P_{err} \geq -36$ dBm0).
 Second, the short term average power of the error signal should preferably exceed the estimated
 power level of the background noise for the error signal by preferably at least about 6 dB ($P_{err} \geq$
 $B_{err} + 6$ dB). Third, the short term average power 124(a) of the near end signal 122(b) is
 15 preferably approximately 3 dB greater than the maximum power level 128(c) of the reference
 signal 126(a) less the estimated ERL ($P_{near} \geq P_{refmax} - ERL + 3$ dB). The adaptation logic 136
 preferably sets a near end speech hangover counter (not shown) when near end speech is
 detected. The hangover counter is used to prevent clipping of near end speech by delaying the
 invocation of the NLP 140 when near end speech is detected. Preferably the hangover counter
 20 is on the order of about 150 msec.

In the described exemplary embodiment, if the maximum power level (P_{refmax}) of the
 reference signal minus the estimated ERL is less than the threshold of hearing (all in dB) neither
 echo cancellation or non-linear processing are invoked. In this instance, the energy level of the
 echo is below the threshold of hearing, typically about -65 to -69 dBm0, so that echo cancellation
 25 and non-linear processing are not required for the current time period. Therefore, the bypass
 estimator 142 sets the bypass cancellation switch 144 in the down position, so as to bypass the
 echo canceller and the NLP and no processing (other than updating the power estimates) is
 performed. Also, if the maximum power level (P_{refmax}) of the reference signal minus the
 estimated ERL is less than the maximum of either the threshold of hearing, or background power
 30 level B_{err} of the error signal minus a predetermined threshold ($P_{refmax} - ERL < \text{threshold of hearing}$
 or ($B_{err} - \text{threshold}$)) neither echo cancellation or non-linear processing are invoked. In this
 instance, the echo is buried in the background noise or below the threshold of hearing, so that
 echo cancellation and non-linear processing are not required for the current time period. In the
 described preferred embodiment the background noise estimate is preferably greater than the
 35 threshold of hearing, such that this is a broader method for setting the bypass cancellation switch.
 The threshold is preferably in the range of about 8-12 dB.

Similarly, if the maximum power level (P_{refmax}) of the reference signal minus the estimated
 ERL is less than the short term average power P_{near} minus a predetermined threshold ($P_{refmax} - ERL$

1 < P_{near} - threshold) neither echo cancellation or non-linear processing are invoked. In this instance, it is highly probable that near end speech is present, and that such speech will likely mask the echo. This method operates in conjunction with the above described techniques for bypassing the echo canceller and NLP. The threshold is preferably in the range of about 8-12 dB.

5 If the NLP contains a real comfort noise generator, i.e., a non-linearity which mutes the incoming signal and injects comfort noise of the appropriate character then a determination that the NLP will be invoked in the absence of filter adaptation allows the adaptive filter to be bypassed or not invoked. This method is used in conjunction with the above methods. If the adaptive filter is not executed then adaptation does not take place, so this method is preferably used only when the

10 echo canceller has converged.

If the bypass cancellation switch 144 is in the down position; the adaptation logic 136 disables the filter adapter 134. Otherwise, for those conditions where the bypass cancellation switch 144 is in the up position so that both adaptation and cancellation may take place, the operation of the preferred adaptation logic 136 proceeds as follows:

15 If the estimated echo return loss enhancement is low (preferably in the range of about 0-9dBm) the adaptation logic 136 enables rapid convergence with an adaptation step size $\alpha = 1/4$. In this instance, the echo canceller is not converged so that rapid adaptation is warranted. However, if near end speech is detected within the hangover period, the adaptation logic 136 either disables adaptation or uses very slow adaptation, preferably an adaptation speed on the

20 order of about one-eighth that used for rapid convergence or an adaptation step size $\alpha = 1/32$. In this case the adaptation logic 136 disables adaptation when the echo canceller is converged. Convergence may be assumed if adaptation has been active for a total of one second after the off hook transition or subsequent to the invocation of the echo canceller. Otherwise if the combined loss (ERL+ERLE) is in the range of about 33-36 dB, the adaptation logic 136 enables slow

25 adaptation (preferably one-eighth the adaptation speed of rapid convergence or an adaptation step size $\alpha = 1/32$). If the combined loss (ERL+ERLE) is in the range of about 23-33 dB, the adaptation logic 136 enables a moderate convergence speed, preferably on the order of about one-fourth the adaptation speed used for rapid convergence or an adaptation step size $\alpha = 1/16$.

Otherwise, one of three preferred adaptation speeds is chosen based on the estimated echo

30 power (P_{refmax} minus the ERL) in relation to the power level of the background noise of the error signal. If the estimated echo power ($P_{refmax} - ERL$) is large compared to the power level of the background noise of the error signal ($P_{refmax} - ERL \geq B_{err} + 24$ dB), rapid adaptation / convergence is enabled with an adaptation step size on the order of about $\alpha = 1/4$. Otherwise, if ($P_{refmax} - ERL \geq B_{err} + 18$ dB) the adaptation speed is reduced to approximately one-half the adaptation speed

35 used for rapid convergence or an adaptation step size on the order of about $\alpha = 1/8$. Otherwise, if ($P_{refmax} - ERL \geq B_{err} + 9$ dB) the adaptation speed is further reduced to approximately one-quarter the adaptation speed used for rapid convergence or an adaptation step size $\alpha = 1/16$.

1 As a further limit on adaptation speed, if echo canceller adaptation has been active for a
sum total of one second since initialization or an off-hook condition then the maximum
adaptation speed is limited to one-fourth the adaptation speed used for rapid convergence
($\alpha=1/16$). Also, if the echo path changes appreciably or if for any reason the estimated ERLE
5 is negative, (which typically occurs when the echo path changes) then the coefficients are cleared
and an adaptation counter is set to zero (the adaptation counter measures the sum total of
adaptation cycles in samples).

The NLP 140 is a two state device. The NLP 140 is either on (applying non-linear
processing) or it is off (applying unity gain). When the NLP 140 is on it tends to stay on, and
10 when the NLP 140 is off it tends to stay off. The NLP 140 is preferably invoked when the bypass
cancellation switch 144 is in the upper position so that adaptation and cancellation are active.
Otherwise, the NLP 140 is not invoked and the NLP 140 is forced into the off state.

Initially, a stateless first NLP decision is created. The decision logic is based on three
decision variables (D1- D3). The decision variable D1 is set if it is likely that the far end is
15 active (i.e. the short term average power 128(a) of the reference signal 126(a) is preferably about
6 dB greater than the power level of the background noise 128(d) of the reference signal), and
the short term average power 128(a) of the reference signal 126(a) minus the estimated ERL is
greater than the estimated short term average power 124(a) of the near end signal 122(b) minus
a small threshold, preferably in the range of about 6 dB. In the preferred embodiment, this is
20 represented by: ($P_{ref} \geq B_{ref} + 6 \text{ dB}$) and ($(P_{ref} - \text{ERL}) \geq (P_{near} - 6 \text{ dB})$). Thus, decision variable D1
attempts to detect far end active speech and high ERL (implying no near end). Preferably,
decision variable D2 is set if the power level of the error signal is on the order of about 9 dB
below the power level of the estimated short term average power 124(a) of the near end signal
122(b) (a condition that is indicative of good short term ERLE). In the preferred embodiment,
25 $P_{err} \leq P_{near} - 9 \text{ dB}$ is used (a short term ERLE of 9 dB). The third decision variable D3 is
preferably set if the combined loss (reference power to error power) is greater than a threshold.
In the preferred embodiment, this is: $P_{err} \leq P_{ref} - t$, where t is preferably initialized to about 6 dB
and preferably increases to about 12 dB after about one second of adaptation. (In other words,
it is only adapted while convergence is enabled).

30 The third decision variable D3 results in more aggressive non linear processing while the
echo canceller is uncovered. Once the echo canceller converges, the NLP 140 can be slightly
less aggressive. The initial stateless decision is set if two of the sub-decisions or control
variables are initially set. The initial decision set implies that the NLP 140 is in a transition state
or remaining on.

35 A NLP state machine (not shown) controls the invocation and termination of NLP 140
in accordance with the detection of near end speech as previously described. The NLP state
machine delays activation of the NLP 140 when near end speech is detected to prevent clipping
the near end speech. In addition, the NLP state machine is sensitive to the near end speech

1 hangover counter (set by the adaptation logic when near end speech is detected) so that activation
of the NLP 140 is further delayed until the near end speech hangover counter is cleared. The
NLP state machine also deactivates the NLP 140. The NLP state machine preferably sets an off
counter when the NLP 140 has been active for a predetermined period of time, preferably about
5 the tail length in msec. The "off" counter is cleared when near end speech is detected and
decremented while non-zero when the NLP is on. The off counter delays termination of NLP
processing when the far end power decreases so as to prevent the reflection of echo stored in the
tail circuit. If the near end speech detector hangover counter is on, the above NLP decision is
overridden and the NLP is forced into the off state.

10 In the preferred embodiment, the NLP 140 may be implemented with a suppressor that
adaptively suppresses down to the background noise level (B_{err}), or a suppressor that suppresses
completely and inserts comfort noise with a spectrum that models the true background noise.

2. Automatic Gain Control

15 In an exemplary embodiment of the present invention, AGC is used to normalize digital
voice samples to ensure that the conversation between the near and far end users is maintained
at an acceptable volume. The described exemplary embodiment of the AGC includes a signal
bypass for the digital voice samples when the gain adjusted digital samples exceeds a
predetermined power level. This approach provides rapid response time to increased power
20 levels by coupling the digital voice samples directly to the output of the AGC until the gain falls
off due to AGC adaptation. Although AGC is described in the context of a signal processing
system for packet voice exchange, those skilled in the art will appreciate that the techniques
described for AGC are likewise suitable for various applications requiring a signal bypass when
the processing of the signal produces undesirable results. Accordingly, the described exemplary
embodiment for AGC in a signal processing system is by way of example only and not by way
25 of limitation.

In an exemplary embodiment, the AGC can be either fully adaptive or have a fixed gain.
Preferably, the AGC supports a fully adaptive operating mode with a range of about -30 dB to
30 dB. A default gain value may be independently established, and is typically 0 dB. If adaptive
gain control is used, the initial gain value is specified by this default gain. The AGC adjusts the
gain factor in accordance with the power level of an input signal. Input signals with a low energy
level are amplified to a comfortable sound level, while high energy signals are attenuated.

35 A block diagram of a preferred embodiment of the AGC is shown in FIG. 8A. A
multiplier 150 applies a gain factor 152 to an input signal 150(a) which is then output to the
media queue 66 of the network VHD via the switchboard 32' (see FIG. 6). The default gain,
typically 0 dB is initially applied to the input signal 150(a). A power estimator 154 estimates the
short term average power 154(a) of the gain adjusted signal 150(b). The short term average
power of the input signal 150(a) is preferably calculated every eight samples, typically every one
ms for a 8 kHz signal. Clipping logic 156 analyzes the short term average power 154(a) to

1 identify gain adjusted signals 150(b) whose amplitudes are greater than a predetermined clipping
threshold. The clipping logic 156 controls an AGC bypass switch 157, which directly connects
the input signal 150(a) to the media queue 66 when the amplitude of the gain adjusted signal
150(b) exceeds the predetermined clipping threshold. The AGC bypass switch 157 remains in
5 the up or bypass position until the AGC adapts so that the amplitude of the gain adjusted signal
150(b) falls below the clipping threshold.

The power estimator 154 also calculates a long term average power 154(b) for the input
signal 150(a), by averaging thirty two short term average power estimates, (i.e. averages thirty
two blocks of eight samples). The long term average power is a moving average which provides
10 significant hangover. A peak tracker 158 utilizes the long term average power 154(b) to calculate
a reference value which gain calculator 160 utilizes to estimate the required adjustment to a gain
factor 152. The gain factor 152 is applied to the input signal 150(a) by the multiplier 150. In the
described exemplary embodiment the peak tracker 158 may preferably be a non-linear filter. The
peak tracker 158 preferably stores a reference value which is dependent upon the last maximum
15 peak. The peak tracker 158 compares the long term average power estimate to the reference
value. FIG. 8B shows the peak tracker output as a function of an input signal, demonstrating that
the reference value that the peak tracker 158 forwards to the gain calculator 160 should
preferably rise quickly if the signal amplitude increases, but decrement slowly if the signal
amplitude decreases. Thus for active voice segments followed by silence, the peak tracker output
20 slowly decreases, so that the gain factor applied to the input signal 150(a) may be slowly
increased. However, for long inactive or silent segments followed by loud or high amplitude
voice segments, the peak tracker output increases rapidly, so that the gain factor applied to the
input signal 150(a) may be quickly decreased.

In the described exemplary embodiment, the peak tracker should be updated when the
25 estimated long term power exceeds the threshold of hearing. Peak tracker inputs include the
current estimated long term power level $a(i)$, the previous long term power estimate, $a(i-1)$, and
the previous peak tracker output $x(i-1)$. In operation, when the long term energy is varying
rapidly, preferably when the previous long term power estimate is on the order of four times
greater than the current long term estimate or vice versa, the peak tracker should go into hangover
30 mode. In hangover mode, the peak tracker should not be updated. The hangover mode prevents
adaptation on impulse noise.

If the long term energy estimate is large compared to the previous peak tracker estimate,
then the peak tracker should adapt rapidly. In this case the current peak tracker output $x(i)$ is
given by:

$$35 \quad x(i) = (7x(i-1) + a(i))/8.$$

where $x(i-1)$ is the previous peak tracker output and $a(i)$ is the current long term power
estimate.

1 If the long term energy is less than the previous peak tracker output, then the peak tracker will adapt slowly. In this case the current peak tracker output $x(i)$ is given by:

$$x(i) = x(i-1) * 255/256.$$

5 Referring to FIG. 9, a preferred embodiment of the gain calculator 160 slowly increments the gain factor 152 for signals below the comfort level of hearing 162 (below minVoice) and decrements the gain for signals above the comfort level of hearing 164 (above MaxVoice). The described exemplary embodiment of the gain calculator 160 decrements the gain factor 152 for signals above the clipping threshold relatively fast, preferably on the order of about 2-4 dB/sec, until the signal has been attenuated approximately 10 dB or the power level of the signal drops to the comfort zone. The gain calculator 160 preferably decrements the gain factor 152 for signals with power levels that are above the comfort level of hearing 164 (MaxVoice) but below the clipping threshold 166 (Clip) relatively slowly, preferably on the order of about 0.1-0.3 dB/sec until the signal has been attenuated approximately 4 dB or the power level of the signal drops to the comfort zone.

15 The gain calculator 160 preferably does not adjust the gain factor 152 for signals with power levels within the comfort zone (between minVoice and MaxVoice), or below the maximum noise power threshold 168 (MaxNoise). The preferred values of MaxNoise, minVoice, MaxVoice, Clip are related to a noise floor 170 and are preferably in 3dB increments. The noise floor is preferably empirically derived by calibrating the host DSP platform with a known load. The noise floor preferably adjustable and is typically within the range of about, -45 to -52 dBm. A MaxNoise value of two corresponds to a power level 6 dB above the noise floor 170, whereas a clip level of nine corresponds to 27 dB above noise floor 170. For signals with power levels below the comfort zone (less than minVoice) but above the maximum noise threshold, the gain calculator 160 preferably increments the gain factor 152 logarithmically at a rate of about 0.1-0.3 dB/sec, until the power level of the signal is within the comfort zone or a gain of approximately 10 dB is reached.

20 In the described exemplary embodiment, the AGC is designed to adapt slowly, although it should adapt fairly quickly if overflow or clipping is detected. From a system point of view, AGC adaptation should be held fixed if the NLP 72 (see FIG. 6) is activated or the VAD 80 (see FIG. 6) determines that voice is inactive. In addition, the AGC is preferably sensitive to the amplitude of received call progress tones. In the described exemplary embodiment, rapid adaptation may be enabled as a function of the actual power level of a received call progress tone such as for example a ring back tone, compared to the power levels set forth in the applicable standards.

35 3. Voice Activity Detector

 In an exemplary embodiment, the VAD, in either the encoder system or the decoder system, can be configured to operate in multiple modes so as to provide system tradeoffs between voice quality and bandwidth requirements. In a first mode, the VAD is always disabled and

1 declares all digital voice samples as active speech. This mode is applicable if the signal
processing system is used over a TDM network, a network which is not congested with traffic,
or when used with PCM (ITU Recommendation G.711 (1988) - Pulse Code Modulation (PCM)
of Voice Frequencies, the contents of which is incorporated herein by reference as if set forth in
5 full) in a PCM bypass mode for supporting data or fax modems.

In a second "transparent" mode, the voice quality is indistinguishable from the first
mode. In transparent mode, the VAD identifies digital voice samples with an energy below the
threshold of hearing as inactive speech. The threshold may be adjustable between -90 and -40
dBm with a default value of -60 dBm. The transparent mode may be used if voice quality is
10 much more important than bandwidth. This may be the case, for example, if a G.711 voice
encoder (or decoder) is used.

In a third "conservative" mode, the VAD identifies low level (but audible) digital voice
samples as inactive, but will be fairly conservative about discarding the digital voice samples.
A low percentage of active speech will be clipped at the expense of slightly higher transmit
15 bandwidth. In the conservative mode, a skilled listener may be able to determine that voice
activity detection and comfort noise generation is being employed. The threshold for the
conservative mode may preferably be adjustable between -65 and -35 dBm with a default value
of -60 dBm.

In a fourth "aggressive" mode, bandwidth is at a premium. The VAD is aggressive about
20 discarding digital voice samples which are declared inactive. This approach will result in speech
being occasionally clipped, but system bandwidth will be vastly improved. The threshold for the
aggressive mode may preferably be adjustable between -60 and -30 dBm with a default value
of -55 dBm.

The transparent mode is typically the default mode when the system is operating with 16
25 bit PCM, companded PCM (G.711) or adaptive differential PCM (ITU Recommendations G.726
(Dec. 1990) - 40, 32, 24, 16 kbit/s Using Low-Delay Code Excited Linear Prediction, and G.727
(Dec. 1990) - 5 -, 4 -, 3 -, and 2 - Sample Embedded Adaptive Differential Pulse Code
Modulation). In these instances, the user is most likely concerned with high quality voice since
a high bit-rate voice encoder (or decoder) has been selected. As such, a high quality VAD should
30 be employed. The transparent mode should also be used for the VAD operating in the decoder
system since bandwidth is not a concern (the VAD in the decoder system is used only to update
the comfort noise parameters). The conservative mode could be used with ITU Recommendation
G.728 (Sept. 1992) - Coding of Speech at 16 kbit/s Using Low-Delay Code Excited Linear
Prediction, G.729, and G.723.1. For systems demanding high bandwidth efficiency, the
35 aggressive mode can be employed as the default mode.

The mechanism in which the VAD detects digital voice samples that do not contain active
speech can be implemented in a variety of ways. One such mechanism entails monitoring the
energy level of the digital voice samples over short periods (where a period length is typically

1 in the range of about 10 to 30 msec). If the energy level exceeds a fixed threshold, the digital voice samples are declared active, otherwise they are declared inactive. The transparent mode can be obtained when the threshold is set to the threshold level of hearing.

5 Alternatively, the threshold level of the VAD can be adaptive and the background noise energy can be tracked. If the energy in the current period is sufficiently larger than the background noise estimate by the comfort noise estimator, the digital voice samples are declared active, otherwise they are declared inactive. The VAD may also freeze the comfort noise estimator or extend the range of active periods (hangover). This type of VAD is used in GSM (European Digital Cellular Telecommunications System; Halfrate Speech Part 6: Voice Activity Detector (VAD) for Half Rate Speech Traffic Channels (GSM 6.42), the contents of which is incorporated herein by reference as if set forth in full) and QCELP (W. Gardner, P. Jacobs, and C. Lee, "QCELP: A Variable Rate Speech Coder for CDMA Digital Cellular," in *Speech and Audio Coding for Wireless and Network Applications*, B.S. atal, V. Cuperman, and A. Gersho (eds.), the contents of which is incorporated herein by reference as if set forth in full).

15 In a VAD utilizing an adaptive threshold level, speech parameters such as the zero crossing rate, spectral tilt, energy and spectral dynamics are measured and compared to stored values for noise. If the parameters differ significantly from the stored values, it is an indication that active speech is present even if the energy level of the digital voice samples is low.

20 When the VAD operates in the conservative or transparent mode, measuring the energy of the digital voice samples can be sufficient for detecting inactive speech. However, the spectral dynamics of the digital voice samples against a fixed threshold may be useful in discriminating between long voice segments with audio spectra and long term background noise. In an exemplary embodiment of a VAD employing spectral analysis, the VAD performs auto-correlations using Itakura or Itakura-Saito distortion to compare long term estimates based on background noise to short term estimates based on a period of digital voice samples. In addition, if supported by the voice encoder, line spectrum pairs (LSPs) can be used to compare long term LSP estimates based on background noise to short terms estimates based on a period of digital voice samples. Alternatively, FFT methods can be used when the spectrum is available from another software module.

30 Preferably, hangover should be applied to the end of active periods of the digital voice samples with active speech. Hangover bridges short inactive segments to ensure that quiet trailing, unvoiced sounds (such as /s/), are classified as active. The amount of hangover can be adjusted according to the mode of operation of the VAD. If a period following a long active period is clearly inactive (i.e., very low energy with a spectrum similar to the measured background noise) the length of the hangover period can be reduced. Generally, a range of about 40 to 300 msec of inactive speech following an active speech burst will be declared active speech due to hangover.

4. Comfort Noise Generator

1 According to industry research the average voice conversation includes as much as sixty percent silence or inactive content so that transmission across the packet based network can be significantly reduced if non-active speech packets are not transmitted across the packet based network. In an exemplary embodiment of the present invention, a comfort noise generator is
5 used to effectively reproduce background noise when non-active speech packets are not received. In the described preferred embodiment, comfort noise is generated as a function of signal characteristics received from a remote source and estimated signal characteristics. In the described exemplary embodiment comfort noise parameters are preferably generated by a comfort noise estimator. The comfort noise parameters may be transmitted from the far end or
10 can be generated by monitoring the energy level and spectral characteristics of the far end noise at the end of active speech (i.e., during the hangover period). Although comfort noise generation is described in the context of a signal processing system for packet voice exchange, those skilled in the art will appreciate that the techniques described for comfort noise generation are likewise suitable for various applications requiring reconstruction of a signal from signal parameters. Accordingly, the described exemplary embodiment for comfort noise generation in a signal processing system for voice applications is by way of example only and not by way of limitation.

A comfort noise generator plays noise. In an exemplary embodiment, a comfort noise generator in accordance with ITU standards G.729 Annex B or G.723.1 Annex A may be used. These standards specify background noise levels and spectral content. Referring to FIG. 6, the
20 VAD 80 in the encoder system determines whether the digital voice samples in the media queue 66 contain active speech. If the VAD 80 determines that the digital voice samples do not contain active speech, then the comfort noise estimator 81 estimates the energy and spectrum of the background noise parameters at the near end to update a long running background noise energy and spectral estimates. These estimates are periodically quantized and transmitted in a SID
25 packet by the comfort noise estimator (usually at the end of a talk spurt and periodically during the ensuing silent segment, or when the background noise parameters change appreciably). The comfort noise estimator 81 should update the long running averages, when necessary, decide when to transmit a SID packet, and quantize and pass the quantized parameters to the packetization engine 78. SID packets should not be sent while the near end telephony device is
30 on-hook, unless they are required to keep the connection between the telephony devices alive. There may be multiple quantization methods depending on the protocol chosen.

In many instances the characterization of spectral content or energy level of the background noise may not be available to the comfort noise generator in the decoder system. For example, SID packets may not be used or the contents of the SID packet may not be specified
35 (see FRF-11). Similarly, the SID packets may only contain an energy estimate, so that estimating some or all of the parameters of the noise in the decoding system may be necessary. Therefore, the comfort noise generator 92 (see FIG. 6) preferably should not be dependent upon SID packets from the far end encoder system for proper operation.

1 In the absence of SID packets, or SID packets containing energy only, the parameters of the background noise at the far end may be estimated by either of two alternative methods. First, the VAD 98 at the voice decoder 96 can be executed in series with the comfort noise estimator 100 to identify silence periods and to estimate the parameters of the background noise during
5 those silence periods. During the identified inactive periods, the digital samples from the voice decoder 96 are used to update the comfort noise parameters of the comfort noise estimator. The far end voice encoder should preferably ensure that a relatively long hangover period is used in order to ensure that there are noise-only digital voice samples which the VAD 98 may identify as inactive speech.

10 Alternatively, in the case of SID packets containing energy levels only, the comfort noise estimate may be updated with the two or three digital voice frames which arrived immediately prior to the SID packet. The far end voice encoder should preferably ensure that at least two or three frames of inactive speech are transmitted before the SID packet is transmitted. This can be realized by extending the hangover period. The comfort noise estimator 100 may then
15 estimate the parameters of the background noise based upon the spectrum and or energy level of these frames. In this alternate approach continuous VAD execution is not required to identify silence periods, so as to further reduce the average bandwidth required for a typical voice channel.

20 Alternatively, if it is unknown whether or not the far end voice encoder supports (sending) SID packets, the decoder system may start with the assumption that SID packets are not being sent, utilizing a VAD to identify silence periods, and then only use the comfort noise parameters contained in the SID packets if and when a SID packet arrives.

25 A preferred embodiment of the comfort noise generator generates comfort noise based upon the energy level of the background noise contained within the SID packets and spectral information derived from the previously decoded inactive speech frames. The described exemplary embodiment (in the decoding system) includes a comfort noise estimator for noise analysis and a comfort noise generator for noise synthesis. Preferably there is an extended hangover period during which the decoded voice samples is primarily inactive before the VAD identifies the signal as being inactive, (changing from speech to noise). Linear Prediction Coding
30 (LPC) coefficients may be used to model the spectral shape of the noise during the hangover period just before the SID packet is received from the VAD. Linear prediction coding models each voice sample as a linear combination of previous samples, that is, as the output of an all-pole IIR filter. Referring to FIG. 10, a noise analyzer 174 determines the LPC coefficients.

35 In the described exemplary embodiment of the comfort noise estimator in the decoding system, a signal buffer 176 receives and buffers decoded voice samples. An energy estimator 177 analyzes the energy level of the samples buffered in the signal buffer 176. The energy estimator 177 compares the estimated energy level of the samples stored in the signal buffer with the energy level provided in the SID packet. Comfort noise estimating is terminated if the energy

1 level estimated for the samples stored in the signal buffer and the energy level provided in the
 SID packet differ by more than a predetermined threshold, preferably on the order of about 6 dB.
 In addition, the energy estimator 177, analyzes the stability of the energy level of the samples
 buffered in the signal buffer. The energy estimator 177 preferably divides the samples stored in
 5 the signal buffer into two groups, (preferably approximately equal halves) and estimates the
 energy level for each group. Comfort noise estimation is preferably terminated if the estimated
 energy levels of the two groups differ by more than a predetermined threshold, preferably on the
 order of about 6 dB. A shaping filter 178 filters the incoming voice samples from the energy
 estimator 177 with a triangular windowing technique. Those of skill in the art will appreciate
 10 that alternative shaping filters such as, for example, a Hamming window, may be used to shape
 the incoming samples.

When a SID packet is received in the decoder system, auto correlation logic 179
 calculates the auto-correlation coefficients of the windowed voice samples. The signal buffer
 176 should preferably be sized to be smaller than the hangover period, to ensure that the auto
 correlation logic 179 computes auto correlation coefficients using only voice samples from the
 15 hangover period. In the described exemplary embodiment, the signal buffer is sized to store on
 the order of about two hundred voice samples (25 msec assuming a sample rate of 8000 Hz).
 Autocorrelation, as is known in the art, involves correlating a signal with itself. A correlation
 function shows how similar two signals are and how long the signals remain similar when one
 is shifted with respect to the other. Random noise is defined to be uncorrelated, that is random
 20 noise is only similar to itself with no shift at all. A shift of one sample results in zero correlation,
 so that the autocorrelation function of random noise is a single sharp spike at shift zero. The
 autocorrelation coefficients are calculated according to the following equation:

$$r(k) = \sum s(n)s(n-k)$$

25 where $k=0 \dots p$ and p is the order of the synthesis filter 188 (see FIG. 11) utilized to
 synthesize the spectral shape of the background noise from the LPC filter coefficients.

Filter logic 180 utilizes the auto correlation coefficients to calculate the LPC filter
 coefficients 180(a) and prediction gain 180(b) using the Levinson-Durbin Recursion method.
 Preferably, the filter logic 180 first preferably applies a white noise correction factor to $r(0)$ to
 30 increase the energy level of $r(0)$ by a predetermined amount. The preferred white noise
 correction factor is on the order of about (257/256) which corresponds to a white noise level of
 approximately 24 dB below the average signal power. The white noise correction factor
 effectively raises the spectral minima so as to reduce the spectral dynamic range of the auto
 correlation coefficients to alleviate ill-conditioning of the Levinson-Durbin recursion. As is
 35 known in the art, the Levinson-Durbin recursion is an algorithm for finding an all-pole IIR filter
 with a prescribed deterministic autocorrelation sequence. The described exemplary embodiment
 preferably utilizes a tenth order (i.e. ten tap) synthesis filter 188. However, a lower order filter
 may be used to realize a reduced complexity comfort noise estimator.

1 The signal buffer 176 should preferably be updated each time the voice decoder is invoked during periods of active speech. Therefore, when there is a transition from speech to noise, the buffer 176 contains the voice samples from the most recent hangover period. The comfort noise estimator should preferably ensure that the LPC filter coefficients is determined using only samples of background noise. If the LPC filter coefficients are determined based on the analysis of active speech samples, the estimated LPC filter coefficients will not give the correct spectrum of the background noise. In the described exemplary embodiment, a hangover period in the range of about 50-250 msec is assumed, and twelve active frames (assuming 5 msec frames) are accumulated before the filter logic 180 calculates new LPC coefficients.

10 In the described exemplary embodiment a comfort noise generator utilizes the power level of the background noise retrieved from processed SID packets and the predicted LPC filter coefficients 180(a) to generate comfort noise in accordance with the following formula:

$$s(n) = e(n) + \sum_{i=1}^M a(i)s(n-i)$$

15 Where M is the order (i.e. the number of taps) of the synthesis filter 188, s(n) is the predicted value of the synthesized noise, a(i) is the ith LPC filter coefficient, s(n-i) are the previous output samples of the synthesis filter and e(n) is a Gaussian excitation signal.

A block diagram of the described exemplary embodiment of the comfort noise generator 182 is shown in FIG. 11. The comfort noise estimator processes SID packets to decode the power level of the current far end background noise. The power level of the background noise is forwarded to a power controller 184. In addition a white noise generator 186 forwards a gaussian signal to the power controller 184. The power controller 184 adjusts the power level of the gaussian signal in accordance with the power level of the background noise and the prediction gain 180(b). The prediction gain is the difference in power level of the input and output of synthesis filter 188. The synthesis filter 188 receives voice samples from the power controller 184 and the LPC filter coefficients calculated by the filter logic 180 (see FIG. 10). The synthesis filter 188 generates a power adjusted signal whose spectral characteristics approximate the spectral shape of the background noise in accordance with the above equation (i.e. sum of the product of the LPC filter coefficients and the previous output samples of the synthesis filter).

5. Voice Encoder/Voice Decoder

30 The purpose of voice compression algorithms is to represent voice with highest efficiency (i.e., highest quality of the reconstructed signal using the least number of bits). Efficient voice compression was made possible by research starting in the 1930's that demonstrated that voice could be characterized by a set of slowly varying parameters that could later be used to reconstruct an approximately matching voice signal. Characteristics of voice perception allow for lossy compression without perceptible loss of quality.

35 Voice compression begins with an analog-to-digital converter that samples the analog voice at an appropriate rate (usually 8,000 samples per second for telephone bandwidth voice) and then represents the amplitude of each sample as a binary code that is transmitted in a serial

1 fashion. In communications systems, this coding scheme is called pulse code modulation (PCM).

When using a uniform (linear) quantizer in which there is uniform separation between amplitude levels. This voice compression algorithm is referred to as "linear", or "linear PCM".

5 Linear PCM is the simplest and most natural method of quantization. The drawback is that the signal-to-noise ratio (SNR) varies with the amplitude of the voice sample. This can be substantially avoided by using non-uniform quantization known as companded PCM..

In companded PCM, the voice sample is compressed to logarithmic scale before transmission, and expanded upon reception. This conversion to logarithmic scale ensures that
10 low-amplitude voice signals are quantized with a minimum loss of fidelity, and the SNR is more uniform across all amplitudes of the voice sample. The process of compressing and expanding the signal is known as "companding" (COMpressing and exPANDING). There exists a worldwide standard for companded PCM defined by the CCITT (the International Telegraph and Telephone Consultative Committee).

15 The CCITT is a Geneva-based division of the International Telecommunications Union (ITU), a New York-based United Nations organization. The CCITT is now formally known as the ITU-T, the telecommunications sector of the ITU, but the term CCITT is still widely used. Among the tasks of the CCITT is the study of technical and operating issues and releasing recommendations on them with a view to standardizing telecommunications on a worldwide
20 basis. A subset of these standards is the G-Series Recommendations, which deal with the subject of transmission systems and media, and digital systems and networks. Since 1972, there have been a number of G-Series Recommendations on speech coding, the earliest being Recommendation G.711. G.711 has the best voice quality of the compression algorithms but the highest bit rate requirement.

25 The ITU-T defined the "first" voice compression algorithm for digital telephony in 1972. It is companded PCM defined in Recommendation G.711. This Recommendation constitutes the principal reference as far as transmission systems are concerned. The basic principle of the G.711 companded PCM algorithm is to compress voice using 8 bits per sample, the voice being sampled at 8 kHz, keeping the telephony bandwidth of 300-3400 Hz. With this combination,
30 each voice channel requires 64 kilobits per second.

Note that when the term PCM is used in digital telephony, it usually refers to the companded PCM specified in Recommendation G.711, and not linear PCM, since most transmission systems transfer data in the companded PCM format. Companded PCM is currently the most common digitization scheme used in telephone networks. Today, nearly every
35 telephone call in North America is encoded at some point along the way using G.711 companded PCM.

1 ITU Recommendation G.726 specifies a multiple-rate ADPCM compression technique for converting 64 kilobit per second companded PCM channels (specified by Recommendation G.711) to and from a 40, 32, 24, or 16 kilobit per second channel. The bit rates of 40, 32, 24, and 16 kilobits per second correspond to 5, 4, 3, and 2 bits per voice sample.

5 ADPCM is a combination of two methods: Adaptive Pulse Code Modulation (APCM), and Differential Pulse Code Modulation (DPCM). Adaptive Pulse Code Modulation can be used in both uniform and non-uniform quantizer systems. It adjusts the step size of the quantizer as the voice samples change, so that variations in amplitude of the voice samples, as well as transitions between voiced and unvoiced segments, can be accommodated. In DPCM systems, 10 the main idea is to quantize the difference between contiguous voice samples. The difference is calculated by subtracting the current voice sample from a signal estimate predicted from previous voice sample. This involves maintaining an adaptive predictor (which is linear, since it only uses first-order functions of past values). The variance of the difference signal results in more efficient quantization (the signal can be compressed coded with fewer bits).

15 The G.726 algorithm reduces the bit rate required to transmit intelligible voice, allowing for more channels. The bit rates of 40, 32, 24, and 16 kilobits per second correspond to compression ratios of 1.6:1, 2:1, 2.67:1, and 4:1 with respect to 64 kilobits per second companded PCM. Both G.711 and G.726 are waveform encoders; they can be used to reduce the bit rate require to transfer any waveform, like voice, and low bit-rate modem signals, while 20 maintaining an acceptable level of quality.

There exists another class of voice encoders, which model the excitation of the vocal tract to reconstruct a waveform that appears very similar when heard by the human ear, although it may be quite different from the original voice signal. These voice encoders, called vocoders, offer greater voice compression while maintaining good voice quality, at the penalty of higher 25 computational complexity and increased delay.

For the reduction in bit rate over G.711, one pays for an increase in computational complexity. Among voice encoders, the G.726 ADPCM algorithm ranks low to medium on a relative scale of complexity, with companded PCM being of the lowest complexity and code-excited linear prediction (CELP) vocoder algorithms being of the highest.

30 The G.726 ADPCM algorithm is a sample-based encoder like the G.711 algorithm, therefore, the algorithmic delay is limited to one sample interval. The CELP algorithms operate on blocks of samples (0.625ms to 30 ms for the ITU coder), so the delay they incur is much greater.

35 The quality of G.726 is best for the two highest bit rates, although it is not as good as that achieved using companded PCM. The quality at 16 kilobits per second is quite poor (a noticeable amount of noise is introduced), and should normally be used only for short periods when it is necessary to conserve network bandwidth (overload situations).

1 The G.726 interface specifies as input to the G.726 encoder (and output to the G.726 decoder) an 8-bit companded PCM sample according to Recommendation G.711. So strictly speaking, the G.726 algorithm is a transcoder, taking log-PCM and converting it to ADPCM, and vice-versa. Upon input of a companded PCM sample, the G.726 encoder converts it to a 14-bit
5 linear PCM representation for intermediate processing. Similarly, the decoder converts an intermediate 14-bit linear PCM value into an 8-bit companded PCM sample before it is output. An extension of the G.726 algorithm was carried out in 1994 to include, as an option, 14-bit linear PCM input signals and output signals. The specification for such a linear interface is given in Annex A of Recommendation G.726.

10 The interface specified by G.726 Annex A bypasses the input and output companded PCM conversions. The effect of removing the companded PCM encoding and decoding is to decrease the coding degradation introduced by the compression and expansion of the linear PCM samples.

15 The algorithm implemented in the described exemplary embodiment can be the version specified in G.726 Annex A, commonly referred to as G.726A, or any other voice compression algorithm known in the art. Among these voice compression algorithms are those standardized for telephony by the ITU-T. Several of these algorithms operate at a sampling rate of 8000 Hz. with different bit rates for transmitting the encoded voice. By way of example, Recommendations G.729 (1996) and G.723.1 (1996) define code excited linear prediction
20 (CELP) algorithms that provide even lower bit rates than G.711 and G.726. G.729 operates at 8 kbps and G.723.1 operates at either 5.3 kbps or 6.3 kbps.

 In an exemplary embodiment, the voice encoder and the voice decoder support one or more voice compression algorithms, including but not limited to, 16 bit PCM (non-standard, and only used for diagnostic purposes); ITU-T standard G.711 at 64 kb/s; G.723.1 at 5.3 kb/s
25 (ACELP) and 6.3 kb/s (MP-MLQ); ITU-T standard G.726 (ADPCM) at 16, 24, 32, and 40 kb/s; ITU-T standard G.727 (Embedded ADPCM) at 16, 24, 32, and 40 kb/s; ITU-T standard G.728 (LD-CELP) at 16 kb/s ; and ITU-T standard G.729 Annex A (CS-ACELP) at 8 kb/s.

 The packetization interval for 16 bit PCM, G.711, G.726, G.727 and G.728 should be a multiple of 5 msec in accordance with industry standards. The packetization interval is the time
30 duration of the digital voice samples that are encapsulated into a single voice packet. The voice encoder (decoder) interval is the time duration in which the voice encoder (decoder) is enabled. The packetization interval should be an integer multiple of the voice encoder (decoder) interval (a frame of digital voice samples). By way of example, G.729 encodes frames containing 80 digital voice samples at 8 kHz which is equivalent to a voice encoder (decoder) interval of 10
35 msec. If two subsequent encoded frames of digital voice sample are collected and transmitted in a single packet, the packetization interval in this case would be 20 msec.

 G.711, G.726, and G.727 encodes digital voice samples on a sample by sample basis. Hence, the minimum voice encoder (decoder) interval is 0.125 msec. This is somewhat of a short

1 voice encoder (decoder) interval, especially if the packetization interval is a multiple of 5 msec. Therefore, a single voice packet will contain 40 frames of digital voice samples. G.728 encodes frames containing 5 digital voice samples (or 0.625 msec). A packetization interval of 5 msec (40 samples) can be supported by 8 frames of digital voice samples. G.723.1 compresses frames
5 containing 240 digital voice samples. The voice encoder (decoder) interval is 30 msec, and the packetization interval should be a multiple of 30 msec.

Packetization intervals which are not multiples of the voice encoder (or decoder) interval can be supported by a change to the packetization engine or the depacketization engine. This may be acceptable for a voice encoder (or decoder) such as G.711 or 16 bit PCM.

10 The G.728 standard may be desirable for some applications. G.728 is used fairly extensively in proprietary voice conferencing situations and it is a good trade-off between bandwidth and quality at a rate of 16 kb/s. Its quality is superior to that of G.729 under many conditions, and it has a much lower rate than G.726 or G.727. However, G.728 is MIPS intensive.

15 Differentiation of various voice encoders (or decoders) may come at a reduced complexity. By way of example, both G.723.1 and G.729 could be modified to reduce complexity, enhance performance, or reduce possible IPR conflicts. Performance may be enhanced by using the voice encoder (or decoder) as an embedded coder. For example, the "core" voice encoder (or decoder) could be G.723.1 operating at 5.3 kb/s with "enhancement"
20 information added to improve the voice quality. The enhancement information may be discarded at the source or at any point in the network, with the quality reverting to that of the "core" voice encoder (or decoder). Embedded coders may be readily implemented since they are based on a given core. Embedded coders are rate scalable, and are well suited for packet based networks. If a higher quality 16 kb/s voice encoder (or decoder) is required, one could use G.723.1 or G.729
25 Annex A at the core, with an extension to scale the rate up to 16 kb/s (or whatever rate was desired).

The configurable parameters for each voice encoder or decoder include the rate at which it operates (if applicable), which companding scheme to use, the packetization interval, and the core rate if the voice encoder (or decoder) is an embedded coder. For G.727, the configuration
30 is in terms of bits/sample. For example EADPCM(5,2) (Embedded ADPCM, G.727) has a bit rate of 40 kb/s (5 bits/sample) with the core information having a rate of 16 kb/s (2 bits/sample).

6. Packetization Engine

35 In an exemplary embodiment, the packetization engine groups voice frames from the voice encoder, and with information from the VAD, creates voice packets in a format appropriate for the packet based network. The two primary voice packet formats are generic voice packets and SID packets. The format of each voice packet is a function of the voice encoder used, the selected packetization interval, and the protocol.

1 Those skilled in the art will readily recognize that the packetization engine could be implemented in the host. However, this may unnecessarily burden the host with configuration and protocol details, and therefore, if a complete self contained signal processing system is desired, then the packetization engine should be operated in the network VHD. Furthermore,
5 there is significant interaction between the voice encoder, the VAD, and the packetization engine, which further promotes the desirability of operating the packetization engine in the network VHD.

 The packetization engine may generate the entire voice packet or just the voice portion of the voice packet. In particular, a fully packetized system with all the protocol headers may be
10 implemented, or alternatively, only the voice portion of the packet will be delivered to the host. By way of example, for VoIP, it is reasonable to create the real-time transport protocol (RTP) encapsulated packet with the packetization engine, but have the remaining transmission control protocol / Internet protocol (TCP/IP) stack residing in the host. In the described exemplary embodiment, the voice packetization functions reside in the packetization engine. The voice
15 packet should be formatted according to the particular standard, although not all headers or all components of the header need to be constructed.

7. Voice Depacketizing Engine / Voice Queue

 In an exemplary embodiment, voice de-packetization and queuing is a real time task which queues the voice packets with a time stamp indicating the arrival time. The voice queue
20 should accurately identify packet arrival time within one msec resolution. Resolution should preferably not be less than the encoding interval of the far end voice encoder. The depacketizing engine should have the capability to process voice packets that arrive out of order, and to dynamically switch between voice encoding methods (i.e. between, for example, G.723.1 and G.711). Voice packets should be queued such that it is easy to identify the voice frame to be
25 released, and easy to determine when voice packets have been lost or discarded en route.

 The voice queue may require significant memory to queue the voice packets. By way of example, if G.711 is used, and the worst case delay variation is 250 msec, the voice queue should be capable of storing up to 500 msec of voice frames. At a data rate of 64 kb/s this translates into 4000 bytes or, or 2K (16 bit) words of storage. Similarly, for 16 bit PCM, 500 msec of voice
30 frames require 4K words. Limiting the amount of memory required may limit the worst case delay variation of 16 bit PCM and possibly G.711. This, however, depends on how the voice frames are queued, and whether dynamic memory allocation is used to allocate the memory for the voice frames. Thus, it is preferable to optimize the memory allocation of the voice queue.

 The voice queue transforms the voice packets into frames of digital voice samples. If the
35 voice packets are at the fundamental encoding interval of the voice frames, then the delay jitter problem is simplified. In an exemplary embodiment, a double voice queue is used. The double voice queue includes a secondary queue which time stamps and temporarily holds the voice packets, and a primary queue which holds the voice packets, time stamps, and sequence numbers.

1 The voice packets in the secondary queue are disassembled before transmission to the primary queue. The secondary queue stores packets in a format specific to the particular protocol, whereas the primary queue stores the packets in a format which is largely independent of the particular protocol.

5 In practice, it is often the case that sequence numbers are included with the voice packets, but not the SID packets, or a sequence number on a SID packet is identical to the sequence number of a previously received voice packet. Similarly, SID packets may or may not contain useful information. For these reasons, it may be useful to have a separate queue for received SID packets.

10 The depacketizing engine is preferably configured to support VoIP, VTOA, VoFR and other proprietary protocols. The voice queue should be memory efficient, while providing the ability to handle dynamically switched voice encoders (at the far end), allow efficient reordering of voice packets (used for VoIP) and properly identify lost packets.

8. Voice Synchronization

15 In an exemplary embodiment, the voice synchronizer analyzes the contents of the voice queue and determines when to release voice frames to the voice decoder, when to play comfort noise, when to perform frame repeats (to cope with lost voice packets or to extend the depth of the voice queue), and when to perform frame deletes (in order to decrease the size of the voice queue). The voice synchronizer manages the asynchronous arrival of voice packets. For those
20 embodiments which are not memory limited, a voice queue with sufficient fixed memory to store the largest possible delay variation is used to process voice packets which arrive asynchronously. Such an embodiment includes sequence numbers to identify the relative timings of the voice packets. The voice synchronizer should ensure that the voice frames from the voice queue can be reconstructed into high quality voice, while minimizing the end-to-end delay. These are
25 competing objectives so the voice synchronizer should be configured to provide system trade-off between voice quality and delay.

Preferably, the voice synchronizer is adaptive rather than fixed based upon the worst case delay variation. This is especially true in cases such as VoIP where the worst case delay variation can be on the order of a few seconds. By way of example, consider a VoIP system with a fixed
30 voice synchronizer based on a worst case delay variation of 300 msec. If the actual delay variation is 280 msec, the signal processing system operates as expected. However, if the actual delay variation is 20 msec, then the end-to-end delay is at least 280 msec greater than required. In this case the voice quality should be acceptable, but the delay would be undesirable. On the other hand, if the delay variation is 330 msec then an underflow condition could exist degrading
35 the voice quality of the signal processing system.

The voice synchronizer performs four primary tasks. First, the voice synchronizer determines when to release the first voice frame of a talk spurt from the far end. Subsequent to the release of the first voice frame, the remaining voice frames are released in an isochronous

1 manner. In an exemplary embodiment, the first voice frame is held for a period of time that is equal or less than the estimated worst case jitter.

Second, the voice synchronizer estimates how long the first voice frame of the talk spurt should be held. If the voice synchronizer underestimates the required "target holding time," jitter
5 buffer underflow will likely result. However, jitter buffer underflow could also occur at the end of a talk spurt, or during a short silence interval. Therefore, SID packets and sequence numbers could be used to identify what caused the jitter buffer underflow, and whether the target holding time should be increased. If the voice synchronizer overestimates the required "target holding time," all voice frames will be held too long causing jitter buffer overflow. In response to jitter
10 buffer overflow, the target holding time should be decreased. In the described exemplary embodiment, the voice synchronizer increases the target holding time rapidly for jitter buffer underflow due to excessive jitter, but decreases the target holding time slowly when holding times are excessive. This approach allows rapid adjustments for voice quality problems while being more forgiving for excess delays of voice packets.

15 Thirdly, the voice synchronizer provides a methodology by which frame repeats and frame deletes are performed within the voice decoder. Estimated jitter is only utilized to determine when to release the first frame of a talk spurt. Therefore, changes in the delay variation during the transmission of a long talk spurt must be independently monitored. On buffer underflow (an indication that delay variation is increasing), the voice synchronizer
20 instructs the lost frame recovery engine to issue voice frames repeats. In particular, the frame repeat command instructs the lost frame recovery engine to utilize the parameters from the previous voice frame to estimate the parameters of the current voice frame. Thus, if frames 1, 2 and 3 are normally transmitted and frame 3 arrives late, frame repeat is issued after frame number 2, and if frame number 3 arrives during this period, it is then transmitted. The sequence
25 would be frames 1,2, a frame repeat of frame 2 and then frame 3. Performing frame repeats causes the delay to increase, which increasing the size of the jitter buffer to cope with increasing delay characteristics during long talk spurts. Frame repeats are also issued to replace voice frames that are lost en route.

Conversely, if the holding time is too large due to decreasing delay variation, the speed
30 at which voice frames are released should be increased. Typically, the target holding time can be adjusted, which automatically compresses the following silent interval. However, during a long talk spurt, it may be necessary to decrease the holding time more rapidly to minimize the excessive end to end delay. This can be accomplished by passing two voice frames to the voice decoder in one decoding interval but only one of the voice frames is transferred to the media
35 queue.

The voice synchronizer must also function under conditions of severe buffer overflow, where the physical memory of the signal processing system is insufficient due to excessive delay

1 variation. When subjected to severe buffer overflow, the voice synchronizer could simply discard voice frames.

The voice synchronizer should operate with or without sequence numbers, time stamps, and SID packets. The voice synchronizer should also operate with voice packets arriving out of order and lost voice packets. In addition, the voice synchronizer preferably provides a variety of configuration parameters which can be specified by the host for optimum performance, including minimum and maximum target holding time. With these two parameters, it is possible to use a fully adaptive jitter buffer by setting the minimum target holding time to zero msec and the maximum target holding time to 500 msec (or the limit imposed due to memory constraints). Although the preferred voice synchronizer is fully adaptive and able to adapt to varying network conditions, those skilled in the art will appreciate that the voice synchronizer can also be maintained at a fixed holding time by setting the minimum and maximum holding times to be equal.

9. Lost Packet Recovery / Frame Deletion

15 In applications where voice is transmitted through a packet based network there are instances where not all of the packets reach the intended destination. The voice packets may either arrive too late to be sequenced properly or may be lost entirely. These losses may be caused by network congestion, delays in processing or a shortage of processing cycles. The packet loss can make the voice difficult to understand or annoying to listen to.

20 Packet recovery refers to methods used to hide the distortions caused by the loss of voice packets. In the described exemplary embodiment, a lost packet recovery engine is implemented whereby missing voice is filled with synthesized voice using the linear predictive coding model of speech. The voice is modelled using the pitch and spectral information from digital voice samples received prior to the lost packets.

25 The lost packet recovery engine, in accordance with an exemplary embodiment, can be completely contained in the decoder system. The algorithm uses previous digital voice samples or a parametric representation thereof, to estimate the contents of lost packets when they occur.

FIG. 12 shows a block diagram of the voice decoder and the lost packet recovery engine. The lost packet recovery engine includes a voice analyzer 192, a voice synthesizer 194 and a selector 196. During periods of no packet loss, the voice analyzer 192 buffers digital voice samples from the voice decoder 96.

30 When a packet loss occurs, the voice analyzer 192 generates voice parameters from the buffered digital voice samples. The voice parameters are used by the voice synthesizer 194 to synthesize voice until the voice decoder 96 receives a voice packet, or a timeout period has elapsed. During voice syntheses, a "packet lost" signal is applied to the selector to output the synthesized voice as digital voice samples to the media queue (not shown).

35 A flowchart of the lost recovery engine algorithm is shown in FIG. 13A. The algorithm is repeated every frame, whether or not there has been a lost packet. Every time the algorithm is performed, a frame of digital voice samples are output. For purposes of explanation, assume a frame length of 5 ms. In this case, the inputs to the lost frame recovery engine are forty

1 samples (5 ms of samples for a sampling rate of 8000 Hz) and a flag specifying whether or not there is voice buffered in the voice analyzer. The output of the lost recovery engine is also forty digital voice samples.

5 First, a check is made to see if there has been a packet loss 191. If so, then a check is made to see if this is the first lost packet in a series of voice packets 193. If it is the first lost packet, then the voice is analysed by calculating the LPC parameters, the pitch, and the voicing decision 195 of the buffered digital samples. If the digital samples are voiced 197, then a residual signal is calculated 199 from the buffered digital voice samples and an excitation signal is created from the residual signal 201. The gain factor for the excitation is set to one. If the speech is unvoiced 197, then the excitation gain factor is determined from a prediction error power calculated during a Levinson-Durbin recursion process 207. Using the parameters determined from the voice analysis, one frame of voice is synthesized 201. Finally, the excitation gain factor is attenuated 203, and the synthesized digital voice samples are output 205.

15 If this is not the first lost packet 193, then a check is made on how many packets have been lost. If the number of lost packets exceeds a threshold 209, then a silence signal is generated and output 211. Otherwise, a frame of digital voice samples are synthesized 201, the excitation gain factor is attenuated 203, and the synthesized digital voice samples are output 205.

20 If there are decoded digital voice samples 191, then a check is performed to see if there was a lost packet the last time the algorithm was executed 213. If so, then one-half of a frame of digital voice samples are synthesized, and overlap-added with the first one-half of the frame of decoded digital voice samples 215. Then, in all cases, the digital voice samples are buffered in the voice analyser and a frame of digital voice samples is output 217.

a. Calculation of LPC Parameters

25 There are two main steps in finding the LPC parameters. First the autocorrelation function $r(i)$ is determined up to $r(M)$ where M is the prediction order. Then the Levinson-Durbin recursion formula is applied to the autocorrelation function to get the LPC parameters.

30 There are several steps involved in calculating the autocorrelation function. The calculations are performed on the most recent buffered digital voice samples. First, a Hamming window is applied to the buffered samples. Then $r(0)$ is calculated and converted to a floating-point format. Next, $r(1)$ to $r(M)$ are calculated and converted to floating-point. Finally, a conditioning factor is applied to $r(0)$ in order to prevent ill conditioning of the autocorrelation matrix for a matrix inversion.

35 The calculation of the autocorrelation function is preferably computationally efficient and makes the best use of fixed point arithmetic. The following equation is used as an estimate of the autocorrelation function from $r(0)$ to $r(M)$:

$$r(i) = \sum_{n=0}^{N-i-1} s[n] \cdot s[n-i]$$

where $s[n]$ is the voice signal and N is the length of the voice window.

1 The value of $r(0)$ is scaled such that it is represented by a mantissa and an exponent. The calculations are performed using 16 bit multiplications and the summed results are stored in a 40-bit register. The mantissa is found by shifting the result left or right such that the most significant bit is in bit 30 of the 40-bit register (where the least significant bit is bit 0) and then
5 keeping bits 16 to 31. The exponent is the number of left shifts required for normalization of the mantissa. The exponent may be negative if a large amplitude signal is present.

 The values calculated for $r(1)$ to $r(M)$ are scaled to use the same exponent as is used for $r(0)$, with the assumption that all values of the autocorrelation function are less than or equal to $r(0)$. This representation in which a series of values are represented with the same exponent is
10 called block floating-point because the whole block of data is represented using the same exponent.

 A conditioning factor of 1025/1024 is applied to $r(0)$ in order to prevent ill conditioning of the R matrix. This factor increases the value of $r(0)$ slightly, which has the effect of making $r(0)$ larger than any other value of $r(i)$. It prevents two rows of the autocorrelation matrix from
15 having equal values or nearly equal values, which would cause ill conditioning of the matrix. When the matrix is ill conditioned, it is difficult to control the numerical precision of results during the Levinson-Durbin recursion.

 Once the autocorrelation values have been calculated, the Levinson-Durbin recursion formula is applied. In the described exemplary embodiment a sixth to tenth order predictor is preferably used.

20 Because of truncation effects caused by the use of fixed point calculations, errors can occur in the calculations when the R matrix is ill conditioned. Although the conditioning factor applied to $r(0)$ eliminates this problem for most cases, there is a numerical stability check implemented in the recursion algorithm. If the magnitude of the reflection coefficient gets greater than or equal to one, then the recursion is terminated, the LPC parameters are set to zero,
25 and the prediction error power is set to $r(0)$.

b. Pitch Period and Voicing Calculation.

 The voicing determination and pitch period calculation are performed using the zero crossing count and autocorrelation calculations. The two operations are combined such that the
30 pitch period is not calculated if the zero crossing count is high since the digital voice samples are classified as unvoiced. FIG. 13B shows a flowchart of the operations performed.

 First the zero crossing count is calculated for a series of digital voice samples 219. The zero crossing count is initialized to zero. The zero crossings are found at a particular point by multiplying the current digital voice sample by the previous digital voice sample and considering
35 the sign of the result. If the sign is negative, then there was a zero crossing and the zero crossing count is incremented. This process is repeated for a number of digital voice samples, and then the zero crossing count is compared to a pre-determined threshold. If the count is above the threshold 221, then the digital voice sample is classified as unvoiced 223. Otherwise, more computations are performed.

Next, if the digital voice samples are not classified as unvoiced, the pitch period is calculated 225. One way to estimate the pitch period in a given segment of speech is to maximize the autocorrelation coefficient over a range of pitch values. This is shown in the equation below:

$$P = \arg \max_p \left(\frac{\sum_{i=0}^{N-p-1} s[i] \cdot s[i+p]}{\sqrt{\sum_{i=0}^{N-p-1} s[i] \cdot s[i]} \cdot \sqrt{\sum_{i=0}^{N-p-1} s[i+p] \cdot s[i+p]}} \right)$$

An approximation to the above equation is used to find the pitch period. First the denominator is approximated by $r(0)$ and the summation limit in the numerator is made independent of p as follows

$$P = \arg \max_p \left(\frac{\sum_{i=0}^{N-P_{\max}-1} s[i] \cdot s[i+p]}{\sum_{i=0}^{N-P_{\max}-1} s[i] \cdot s[i]} \right)$$

where p is the set of integers greater than or equal to P_{\min} (preferably on the order of about 20 samples) and less than or equal to P_{\max} (preferably on the order of about 130 samples). Next, the denominator is removed since it does not depend on p

$$P = \arg \max_p \left(\sum_{i=0}^{N-P_{\max}-1} s[i] \cdot s[i+p] \right)$$

Finally, the speech arrays are indexed such that the most recent samples are emphasized in the estimation of the pitch

$$P = \arg \max_p \left(\sum_{i=0}^{N-P_{\max}-1} s[N-1-i] \cdot s[N-1-i-p] \right)$$

1 This change improves the performance when the pitch is changing in the voice segment under analysis.

5 When the above equation is applied, a further savings in computations is made by searching only odd values of p . Once the maximum value has been determined, a finer search is implemented by searching the two even values of p on either side of the maximum. Although this search procedure is non-optimal, it normally works well because the autocorrelation function is quite smooth for voiced segments.

10 Once the pitch period has been calculated, the voicing decision is made using the maximum autocorrelation value 227. If the result is greater than 0.38 times $r(0)$ then the digital samples are classified as voiced 229. Otherwise it is classified as unvoiced 223.

c. Excitation Signal Calculation.

15 For voiced samples, the excitation signal for voice synthesis is derived by applying the following equation to the buffered digital voice samples:

$$e[n] = s[n] - \sum_{i=1}^M a_i \cdot s[n-i]$$

20 d. Excitation Gain Factor for Unvoiced Speech.

25 For unvoiced samples, the excitation signal for voice synthesis is a white Gaussian noise sequence with a variance of one quarter. In order to synthesize the voice at the correct level, a gain factor is derived from the prediction error power derived during the Levinson-Durbin recursion algorithm. The prediction error power level gives the power level of the excitation signal that will produce a synthesized voice with power level $r(0)$. Since a gain level is desired rather than a power level, the square root of the prediction error power level is calculated. To make up for the fact that the Gaussian noise has a power of one quarter, the gain is multiplied by a factor of two.

e. Voiced Synthesis.

30 The voiced synthesis is performed every time there is a lost voiced packet and also for the first decoded voiced packet after a series of lost packets. FIG. 13C shows the steps performed in the synthesis of voice.

35 First, the excitation signal is generated. If the samples are voiced 231, then the excitation is generated from the residual signal 233. A residual buffer in the voice analyzer containing the residual signal is modulo addressed such that the excitation signal is equal to repetitions of the past residual signal at the pitch period P :

$$e(n) = \begin{cases} e(n-P) & \text{for } n < P \\ e(n-2P) & \text{for } P \leq n < 2P \\ e(n-3P) & \text{for } 2P \leq n < 3P \\ \dots \end{cases}$$

If the value of P is less than the number of samples to be synthesized, then the excitation signal is repeated more than once. If P is greater than the number of samples to be generated, then less than one pitch period is contained in the excitation. In both cases the algorithm keeps track of the last index into the excitation buffer such that it can begin addressing at the correct point for the next time voice synthesis is required.

If the samples are unvoiced, then a series of Gaussian noise samples are generated. Every sample is produced by the addition of twelve uniformly distributed random numbers. Uniformly distributed samples are generated using the linear congruential method (Knuth, 9) as shown by the following equation

$$X_{n+1} = (aX_n + c) \bmod m$$

where a is set to 32763, c to zero, and m to 65536. Knuth, D. "the art of Computer Programming, Volume 2, Seminumerical Algorithms," Addison Wesley, 1969. The initial value of X_n is equal to 29. The sequence of random numbers repeats every 16384 values, which is the maximum period for the chosen value of m when c is equal to zero. By choosing c not equal to zero the period of repetition could be increased to 65536, but 16384 is sufficient for voice synthesis. The longest segment of voice synthesized by the algorithm is twelve blocks of forty samples, which requires only 5760 uniformly distributed samples. By setting c to zero, the number of operations to calculate the Gaussian random sample is reduced by one quarter.

After the excitation has been constructed, the excitation gain factor is applied to each sample. Finally, the synthesis filter is applied to the excitation to generate the synthetic voice.

f. Overlap-Add Calculation.

The overlap-add process is performed when the first good packet arrives after one or more lost packets. The overlap-add reduces the discontinuity between the end of the synthesized voice and the beginning of the decoded voice. To overlap the two voice signals, additional digital voice samples (equal to one-half of a frame) is synthesized and averaged with the first one-half frame of the decoded voice packet. The synthesized voice is multiplied by a down-sloping linear ramp and the decoded voice is multiplied by an up-sloping linear ramp. Then the two signals are added together.

10. DTMF

DTMF (dual-tone, multi-frequency) tones are signaling tones carried within the audio band. A dual tone signal is represented by two sinusoidal signals whose frequencies are separated in bandwidth and which are uncorrelated to avoid false tone detection. A DTMF signal includes one of four tones, each having a frequency in a high frequency band, and one of four tones, each having a frequency in a low frequency band. The frequencies used for DTMF encoding and detection are defined by the ITU and are widely accepted around the world.

In an exemplary embodiment of the present invention, DTMF detection is performed by sampling only a portion of each voice frame. This approach results in improved overall system efficiency by reducing the complexity (MIPS) of the DTMF detection. Although the DTMF is described in the context of a signal processing system for packet voice exchange, those skilled in the art will appreciate that the techniques described for DMTF are likewise suitable for various applications requiring signal detection by sampling a portion of the signal. Accordingly, the described exemplary embodiment for DTMF in a signal processing system is by way of example only and not by way of limitation.

There are numerous problems involved with the transmission of DTMF in band over a packet based network. For example, lossy voice compression may distort a valid DTMF tone or sequence into an invalid tone or sequence. Also voice packet losses of digital voice samples may corrupt DTMF sequences and delay variation (jitter) may corrupt the DTMF timing information and lead to lost digits. The severity of the various problems depends on the particular voice decoder, the voice decoder rate, the voice packet loss rate, the delay variation, and the particular implementation of the signal processing system. For applications such as VoIP with potentially significant delay variation, high voice packet loss rates, and low digital voice sample rate (if G.723.1 is used), packet tone exchange is desirable. Packet tone exchange is also desirable for VoFR (FRF-11, class 2). Thus, proper detection and out of band transfer via the packet based network is useful.

The ITU and Bellcore have promulgated various standards for DTMF detectors. The described exemplary DTMF detector preferably complies with ITU-T Standard Q.24 (for DTMF digit reception) and Bellcore GR-506-Core, TR-TSY-000181, TR-TSY-000762 and TR-TSY-000763, the contents of which are hereby incorporated by reference as though set forth in full herein. These standards involve various criteria, such as frequency distortion allowance, twist allowance, noise immunity, guard time, talk-down, talk-off, acceptable signal to noise ratio, and dynamic range, etc. which are summarized in the table below.

The distortion allowance criteria specifies that a DTMF detector should detect a transmitted signal that has a frequency distortion of less than 1.5% and should not detect any DTMF signals that have frequency distortion of more than 3.5%. The term "twist" refers to the difference, in decibels, between the amplitude of the strongest key pad column tone and the

1 amplitude of the strongest key pad row tone. For example, the Bellcore standard requires the
twist to be between -8 and +4 dBm. The noise immunity criteria requires that if the signal has
a signal to noise ratio (SNR) greater than certain decibels, then the DTMF detector is required
to not miss the signal, i.e., is required to detect the signal. Different standards have different
5 SNR requirements, which usually range from 12 to 24 decibels. The guard time check criteria
requires that if a tone has a duration greater than 40 milliseconds, the DTMF detector is required
to detect the tone, whereas if the tone has a duration less than 23 milliseconds, the DTMF
detector is required to not detect the tone. Similarly, the DTMF detector is required to accept
interdigit intervals which are greater than or equal to 40 milliseconds. Alternate embodiments
10 of the present invention readily provide for compliance with other telecommunication standards
such as EIA-464B, and JJ-20.12.

Referring to FIG. 14 the DTMF detector 76 processes the 64kb/s pulse code modulated
(PCM) signal, i.e., digital voice samples 76(a) buffered in the media queue (not shown). The
input to the DTMF detector 76 should preferably be sampled at a rate that is at least higher than
15 approximately 4 kHz or twice the highest frequency of a DTMF tone. If the incoming signal
(i.e., digital voice samples) is sampled at a rate that is greater than 4 kHz (i.e. Nyquist for highest
frequency DTMF tone) the signal may immediately be downsampled so as to reduce the
complexity of subsequent processing. The signal may be downsampled by filtering and
discarding samples.

20 A block diagram of an exemplary embodiment of the invention is shown in FIG. 14. The
described exemplary embodiment includes a system for processing the upper frequency band
tones and a substantially similar system for processing the lower frequency band tones. A filter
210 and sampler 212 may be used to down-sample the incoming signal. In the described
exemplary embodiment, the sampling rate is 8 kHz and the front end filter 210 and sampler 212
25 do not down-sample the incoming signal. The output of the sampler 212 is filtered by two
bandpass filters ($H_h(z)$ 214 and $G_h(z)$ 216) for the upper frequency band and two bandpass filters
($H_l(z)$ 218 and $G_l(z)$ 220) for the lower frequency band and down-sampled by samplers 222, 224
for the upper frequency band and 226, 228 for the lower frequency band. The bandpass filters
(214, 216 and 218, 220) for each frequency band are designed using a pair of lowpass filters, one
filter $H(z)$ which multiplies the down-sampled signal by $\cos(2\pi f_h nT)$ and the other filter $G(z)$
30 which multiplies the down-sampled signal by $\sin(2\pi f_h nT)$ (where $T = 1/f_s$ where f_s is the
sampling frequency after the front end down-sampling by the filter 210 and the sampler 212).

In the described exemplary embodiment, the bandpass filters (214, 216 and 218, 220) are
executed every eight samples and the outputs (214a, 216a and 218a, 220a) of the bandpass filters
(214, 216 and 218, 220) are down-sampled by samplers 222, 224 and 226, 228 at a ratio of eight
35 to one. The combination of down-sampling is selected so as to optimize the performance of a
particular DSP in use and preferably provides a sample approximately every msec or a 1 kbs
signal. Down-sampled signals in the upper and lower frequency bands respectively are real
signals. In the upper frequency band, a multiplier 230 multiplies the output of sampler 224 by

1 the square root of minus one (i.e. j) 232. A summer 234 then adds the output of downsampler 222 with the imaginary signal 230(a). Similarly, in the lower frequency band, a multiplier 236 multiplies the output of downsampler 228 by the square root of minus one (i.e. j) 238. A summer 240 then adds the output of downsampler 226 with the imaginary signal 236(a). Combined signals $x_h(t)$ 234(a) and $x_l(t)$ 240(a) at the output of the summers 234, 240 are complex signals. It will be appreciated by one of skill in the art that the function of the bandpass filters can be accomplished by alternative finite impulse response filters or structures such as windowing followed by DFT processing.

10 If a single frequency is present within the bands defined by the bandpass filters, the combined complex signals $x_h(t)$ and $x_l(t)$ will be constant envelope (complex) signals. Short term power estimator 242 and 244 measure the power of $x_h(t)$ and $x_l(t)$ respectively and compare the estimated power levels of $x_h(t)$ and $x_l(t)$ with the requirements promulgated in ITU-T Q.24. In the described exemplary embodiment, the upper band processing is first executed to determine if the power level within the upper band complies with the thresholds set forth in the ITU-T Q.24 recommendations. If the power within the upper band does not comply with the ITU-T recommendations the signal is not a DTMF tone and processing is terminated. If the power within the upper band complies with the ITU-T Q.24 standard, the lower band is processed. A twist estimator 246 compares the power in the upper band and the lower band to determine if the twist (defined as the ratio of the power in the lower band and the power in the upper band) is within an acceptable range as defined by the ITU-T recommendations. If the ratio of the power within the upper band and lower band is not within the bounds defined by the standards, a DTMF tone is not present and processing is terminated.

25 If the ratio of the power within the upper band and lower band complies with the thresholds defined by the ITU-T Q.24 and Bellcore GR-506-Core, TR-TSY-000181, TR-TSY-000762 and TR-TSY-000763 standards, the frequency of the upper band signal $x_h(t)$ and the frequency of the lower band signal $x_l(t)$ are estimated. Because of the duration of the input signal (one msec), conventional frequency estimation techniques such as counting zero crossings may not sufficiently resolve the input frequency. Therefore, differential detectors 248 and 250 are used to estimate the frequency of the upper band signal $x_h(t)$ and the lower band signal $x_l(t)$ respectively. The differential detectors 248 and 250 estimate the phase variation of the input signal over a given time range. Advantageously, the accuracy of estimation is substantially insensitive to the period over which the estimation is performed. With respect to upper band input $x_h(n)$, (and assuming $x_h(n)$ is a sinusoid of frequency f_i) the differential detector 248 computes:

$$y_h(n) = x_h(n)x_h(n-1)^*e^{-j2\pi f_{mid}}$$

where f_{mid} is the mean of the frequencies in the upper band or lower band and superscript* implies complex conjugation. Then,

$$y_h(n) = e(j2\pi f_i n) e(-j2\pi f_i(n-1)) e(-j2\pi f_{mid}) = e(j2\pi(f_i - f_{mid}))$$

which is a constant, independent of n . Arctan functions 252 and 254 each takes the complex input and computes the angle of the above complex value that uniquely identifies the frequency present in the upper and lower bands. In operation $\text{atan2}(\sin(2\pi(f_i - f_{mid})), \cos(2\pi(f_i - f_{mid})))$ returns to within a scaling factor the frequency difference $f_i - f_{mid}$. Those skilled in the art will appreciate that various algorithms, such as a frequency discriminator, could be used to estimate the frequency of the DTMF tone by calculating the phase variation of the input signal over a given time period.

Having estimated the frequency components of the upper band and lower band, the DTMF detector analyzes the upper band and lower band signals to determine whether a DTMF digit is present in the incoming signals and if so which digit. Frequency calculators 256 and 258 compute a mean and variance of the frequency deviation over the entire window of frequency estimates to identify valid DTMF tones in the presence of background noise or speech that resembles a DTMF tone. In the described exemplary embodiment, if the mean of the frequency estimates over the window is within acceptable limits, preferably less than $\pm 2.8\%$ for the lowband and $\pm 2.5\%$ for the highband the variance is computed. If the variance is less than a predetermined threshold, preferably on the order of about 1464 Hz^2 (i.e. standard deviation of 38.2 Hz) the frequency is declared valid. Referring to FIG. 14A, DTMF control logic 259 compares the frequency identified for the upper and lower bands to the frequency pairs identified in the ITU-T recommendations to identify the digit. The DTMF control logic 259 forwards a tone detection flag 259(b) to a state machine 260. The state machine 260 analyzes the time sequence of events and compares the tone on and tone off periods for a given tone to the ITU-T recommendations to determine whether a valid dual tone is present. In the described exemplary embodiment the total window size is preferably 5 msec so that a DTMF detection decision is performed every 5 msec.

In the context of an exemplary embodiment of the voice mode, the DTMF detector is operating in the packet tone exchange along with a voice encoder operating under the packet voice exchange, which allows for simplification of DTMF detection processing. Most voice encoders operate at a particular frame size (the number of voice samples or time in msec over which voice is compressed). For example, the frame size for ITU-T standard G.723.1 is 30 msec. For ITU-T standard G.729 the frame size is 10 msec. In addition, many packet voice systems group multiple output frames from a particular voice encoder into a network cell or packet. To prevent leakage through the voice path, the described exemplary embodiment delays DTMF detection until the last frame of speech is processed before a full packet is constructed. Therefore, for transmissions in accordance with the G.723.1 standard and a single output frame placed into a packet, DTMF detection may be invoked every 30 msec (synchronous with the end of the frame). Under the G.729 standard with two voice encoder frames placed into a single packet, DTMF detection or decision may be delayed until the end of the second voice frame within a packet is processed.

1 In the described exemplary embodiment, the DTMF detector is inherently stateless, so
that detection of DTMF tones within the second 5 msec DTMF block of a voice encoder frame
doesn't depend on DTMF detector processing of the first 5 msec block of that frame. If the delay
in DTMF detection is greater than or equal to twice the DTMF detector block size, the processing
5 required for DTMF detection can be further simplified. For example, the instructions required
to perform DTMF detection may be reduced by 50% for a voice encoder frame size of 10 msec
and a DTMF detector frame size of 5 msec. The ITU-T Q.24 standard requires DTMF tones to
have a minimum duration of 23 msec and an inter-digit interval of 40 msec. Therefore, by way
of example, a valid DTMF tone may be detected within a given 10 msec frame by only analyzing
10 the second 5 msec interval of that frame. Referring to FIG. 14A, in the described exemplary
embodiment, the DTMF control logic 259 analyzes DTMF detector output 76(a) and selectively
enables DTMF detection analysis 259(a) for a current frame segment, as a function of whether
a valid dual tone was detected in previous and future frame segments. For example, if a DTMF
tone was not detected in the previous frame and if DTMF is not present in the second 5 msec
interval of the current frame, then the first 5 msec block need not be processed so that DTMF
15 detection processing is reduced by 50%. Similar savings may be realized if the previous frame
did contain a DTMF (if the DTMF is still present in the second 5 msec portion it is most likely
that it was on in the first 5 msec portion). This method is easily extended to the case of longer
delays (30 msec for G.723.1 or 20-40 msec for G.729 and packetization intervals from 2-4 or
more). It may be necessary to search more than one 5 msec period out of the longer interval, but
only a subset is necessary.

20 DTMF events are preferably reported to the host. This allows the host, for example, to
convert the DTMF sequence of keys to a destination address. It will, therefore, allow the host
to support call routing via DTMF.

25 Depending on the protocol, the packet tone exchange may support muting of the received
digital voice samples, or discarding voice frames when DTMF is detected. In addition, to avoid
DTMF leakage into the voice path, the voice packets may be queued (but not released) in the
encoder system when DTMF is pre-detected. DTMF is pre-detected through a combination of
DTMF decisions and state machine processing. The DTMF detector will make a decision (i.e.
is there DTMF present) every five msec. A state machine 260 analyzes the history of a given
30 DTMF tone to determine the current duration of a given tone so as to estimate how long the tone
will likely continue. If the detection was false (invalid), the voice packets are ultimately released,
otherwise they are discarded. This will manifest itself as occasional jitter when DTMF is falsely
pre-detected. It will be appreciated by one of skill in the art that tone packetization can
alternatively be accomplished through compliance with various industry standards such as for
example, the Frame Relay Forum (FRF-11) standard, the voice over atm standard ITU-TI.363.2,
35 and IETF-draft-avt-tone-04, RTP Payload for DTMF Digits for Telephony Tones and Telephony
Signals, the contents of which are hereby incorporated by reference as though set forth in full.

Software to route calls via DTMF can be resident on the host or within the signal

1 processing system. Essentially, the packet tone exchange traps DTMF tones and reports them
to the host or a higher layer. In an exemplary embodiment, the packet tone exchange will
generate dial tone when an off-hook condition is detected. Once a DTMF digit is detected, the
dial tone is terminated. The packet tone exchange may also have to play ringing tone back to the
5 near end user (when the far end phone is being rung), and a busy tone if the far end phone is
unavailable. Other tones may also need to be supported to indicate all circuits are busy, or an
invalid sequence of DTMF digits were entered.

10 11. Call Progress Tone Detection

Telephone systems provide users with feedback about what they are doing in order to
simplify operation and reduce calling errors. This information can be in the form of lights,
displays, or ringing, but is most often audible tones heard on the phone line. These tones are
15 generally referred to as call progress tones, as they indicate what is happening to dialed phone
calls. Conditions like busy line, ringing called party, bad number, and others each have
distinctive tone frequencies and cadences assigned them for which some standards have been
established. A call progress tone signal includes one of four tones. The frequencies used for call
progress tone encoding and detection, namely 350, 440, 480, and 620 Hz, are defined by the
international telecommunication union and are widely accepted around the world. The relatively
20 narrow frequency separation between tones, 40Hz in one instance complicates the detection of
individual tones. In addition, the duration or cadence of a given tone is used to identify alternate
conditions.

An exemplary embodiment of the call progress tone detector analyzes the spectral
25 (frequency) characteristics of an incoming telephony voice-band signal and generates a tone
detection flag as a function of the spectral analysis. The temporal (time) characteristics of the
tone detection flags are then analyzed to detect call progress tone signals. The call progress tone
detector then forwards the call progress tone signal to the packetization engine to be packetized
and transmitted across the packet based network. Although the call progress tone detector is
described in the context of a signal processing system for packet voice exchange, those skilled
30 in the art will appreciate that the techniques described for call progress tone detection are
likewise suitable for various applications requiring signal detection by analyzing spectral or
temporal characteristics of the signal. Accordingly, the described exemplary embodiment for
precision tone detection in a signal processing system is by way of example only and not by way
of limitation.

35 The described exemplary embodiment preferably includes a call progress tone detector
that operates in accordance with industry standards for the power level (Bellcore SR3004-CPE
Testing Guidelines; Type III Testing) and cadence (Bellcore GR506-Core and Bellcore LSSGR
Signaling For Analog Interface, Call Purpose Signals) of a call progress tone. The call progress

1 tone detector interfaces with the media queue to detect incoming call progress tone signals such as dial tone, re-order tone, audible ringing and line busy or hook status. The problem of call progress tone signaling and detection is a common telephony problem. In the context of packet voice systems in accordance with an exemplary embodiment of the present invention, telephony
5 devices are coupled to a signal processing system which, for the purposes of explanation, is operating in a network gateway to support the exchange of voice between a traditional circuit switched network and a packet based network. In addition, the signal processing system operating on network gateways also supports the exchange of voice between the packet based network and a number of telephony devices.

10 Referring to FIG. 15 the call progress tone detector 264 continuously monitors the media queue 66 of the voice encoder system. Typically the call progress tone detector 264 is invoked every ten msec. Thus, for an incoming signal sampled at a rate of 8 kHz, the preferred call progress tone detector operates on blocks of eighty samples. The call progress tone detector 264 includes a signal processor 266 which analyzes the spectral characteristics of the samples
15 buffered in the media queue 66. The signal processor 266 performs anti-aliasing, decimation, bandpass filtering, and frequency calculations to determine if a tone at a given frequency is present. A cadence processor 268 analyzes the temporal characteristics of the processed tones by computing the on and off periods of the incoming signal. If the cadence processor 268 detects a call progress tone for an acceptable on and off period in accordance with the Bellcore GR506-Core standard, a "Tone Detection Event" will be generated.

20 A block diagram for an exemplary embodiment of the signal processor 266 is shown in FIG. 16. An anti-aliasing low pass filter 270, with a cutoff frequency of preferably about 666Hz, filters the samples buffered in the media queue so as to remove frequency components above the highest call progress tone frequency, i.e. 660 Hz. A down sampler 272 is coupled to the output of the low pass filter 270. Assuming an 8 kHz input signal, the down sampler 272 preferably
25 decimates the low pass filtered signal at a ratio of six:one (which avoids aliasing due to under sampling). The output 272(a) of down sampler 272 is filtered by eight bandpass filters (274, 276, 278, 280, 282, 284, 286 and 288), (i.e. two filters for each call progress tone frequency). The decimation effectively increases the separation between tones, so as to relax the roll-off requirements (i.e. reduce the number of filter coefficients) of the bandpass filters 274-288 which
30 simplifies the identification of individual tones. In the described exemplary embodiment, the bandpass filters for each call progress tone 274-288 are designed using a pair of lowpass filters, one filter which multiplies the down sampled signal by $\cos(2\pi f_n nT)$ and the other filter which multiplies the down sampled signal by $\sin(2\pi f_n nT)$ (where $T = 1/f_s$ where f_s is the sampling frequency after the decimation by the down sampler 272. The outputs of the band pass filters are real signals. Multipliers (290, 292, 294 and 296) multiply the outputs of filters (276, 280, 284
35 and 288) respectively by the square root of minus one (i.e. j) 298 to generate an imaginary component. Summers (300, 302, 304 and 306) then add the outputs of filters (274, 278, 282 and 286) with the imaginary components (290a, 292a, 294a and 296a) respectively. The combined signals are complex signals. It will be appreciated by one of skill in the art that the function of

1 the bandpass filters (274-288) can be accomplished by alternative finite impulse response filters or structures such as windowing followed by DFT processing.

5 Power estimators (308, 310, 312 and 314) estimate the short term average power of the combined complex signals (300a, 302a, 304a and 306a) for comparison to power thresholds determined in accordance with the recommended standard (Bellcore SR3004-CPE Testing Guidelines For Type III Testing). The power estimators 308-312 forward an indication to power state machines (316, 318, 320 and 322) respectively which monitor the estimated power levels within each of the call progress tone frequency bands. Referring to FIG. 17, the power state machine is a three state device, including a disarm state 324, an arm state 326, and a power on state 328. As is known in the art, the state of a power state machine depends on the previous state and the new input. For example, if an incoming signal is initially silent, the power estimator 308 would forward an indication to the power state machine 316 that the power level is less than the predetermined threshold. The power state machine would be off, and disarmed. If the power estimator 308 next detects an incoming signal whose power level is greater than the predetermined threshold, the power estimator forwards an indication to the power state machine 316 indicating that the power level is greater than the predetermined threshold for the given incoming signal. The power state machine 316 switches to the off but armed state. If the next input is again above the predetermined threshold, the power estimator 308 forwards an indication to the power state machine 316 indicating that the power level is greater than the predetermined threshold for the given incoming signal. The power state machine 316 now toggles to the on and armed state. The power state machine 316 substantially reduces or eliminates false detections due to glitches, white noise or other signal anomalies.

25 Turning back to FIG. 16, when the power state machine is set to the on state, frequency calculators (330, 332, 334 and 336) estimate the frequency of the combined complex signals. The frequency calculators (330-336), utilize a differential detection algorithm to estimate the frequency within each of the four call progress tone bands. The frequency calculators (330-336) estimate the phase variation of the input signal over a given time range. Advantageously, the accuracy of the estimation is substantially insensitive to the period over which the estimation is performed. Assuming a sinusoidal input $x(n)$ of frequency f_i the frequency calculator computes:

$$30 \quad y(n) = x(n)x(n-1)^*e(-j2\pi f_{mid})$$

where f_{mid} is the mean of the frequencies within the given call progress tone group and superscript* implies complex conjugation. Then,

$$35 \quad \begin{aligned} y(n) &= e(j2\pi f_i n) e(-j2\pi f_i(n-1))e(-j2\pi f_{mid}) \\ &= e(j2\pi(f_i - f_{mid})) \end{aligned}$$

1 which is a constant, independent of n . The frequency calculators (330-336) then invoke
 an arctan function that takes the complex signal and computes the angle of the above complex
 value that identifies the frequency present within the given call progress tone band. In operation
 $\text{atan2}(\sin(2\pi(f_i - f_{\text{mid}})), \cos(2\pi(f_i - f_{\text{mid}})))$ returns to within a scaling factor the frequency difference
 5 $f_i - f_{\text{mid}}$. Those skilled in the art will appreciate that various algorithms, such as a frequency
 discriminator, could be used to estimate the frequency of the call progress tone by calculating the
 phase variation of the input signal over a given time period.

The frequency calculators (330-336) compute the mean of the frequency deviation over
 the entire 10 msec window of frequency estimates to identify valid call progress tones in the
 10 presence of background noise or speech that resembles a call progress tone. If the mean of the
 frequency estimates over the window is within acceptable limits as summarized by the table
 below, a tone on flag is forwarded to the cadence processor. The frequency calculators (330-336)
 are preferably only invoked if the power state machine is in the on state thereby reducing the
 processor loading (i.e. fewer MIPS) when a call progress tone signal is not present.

Tone	Frequency One / Mean	Frequency Two / Mean
Dial Tone	350 Hz / 2 Hz	440 Hz / 3 Hz
Busy	480 Hz / 7 Hz	620 Hz / 9 Hz
Re-order	480 Hz / 7 Hz	620 Hz / 9 Hz
20 Audible Ringing	440 Hz / 7 Hz	480 Hz / 7 Hz

Referring to FIG. 18A, the signal processor 266 forwards a tone on / tone off indication
 to the cadence processor 268 which considers the time sequence of events to determine whether
 25 a call progress tone is present. Referring to FIG. 18, in the described exemplary embodiment,
 the cadence processor 268 preferably comprises a four state, cadence state machine 340,
 including a cadence tone off state 342, a cadence tone on state 344, a cadence tone arm state 346
 and an idle state 348 (see FIG. 18). The state of the cadence state machine 340 depends on the
 previous state and the new input. For example, if an incoming signal is initially silent, the signal
 30 processor would forward a tone off indication to the cadence state machine 340. The cadence
 state machine 340 would be set to a cadence tone off and disarmed state. If the signal processor
 next detects a valid tone, the signal processor forwards a tone on indication to the cadence state
 machine 340. The cadence state machine 340 switches to a cadence off but armed state.
 Referring to FIG. 18A, the cadence state machine 340 preferably invokes a counter 350 that
 monitors the duration of the tone indication. If the next input is again a valid call progress tone,
 35 the signal processor forwards a tone on indication to the cadence state machine 340. The cadence
 state machine 340 now toggles to the cadence tone on and cadence tone armed state. The
 cadence state machine 340 would remain in the cadence tone on state until receiving two
 consecutive tone off indications from the signal processor at which time the cadence state
 machine 340 sends a tone off indication to the counter 350. The counter 350, resets and

forwards the duration of the on tone to cadence logic 352. The cadence processor 268 similarly estimates the duration of the off tone, which the cadence logic 352 utilizes to determine whether a particular tone is present by comparing the duration of the on tone, off tone signal pair at a given tone frequency to the tone plan recommended in industry standard as summarized in the table below.

Tone	Duration of Tone On / Tolerance	Duration of Tone Off / Tolerance
Dial Tone	Continuous On	No Off Tone
Busy	500 msec / (+/-50 msec)	500 msec / (+/-50 msec)
Re-order	250 msec / (+/-25 msec)	200 msec / (+/-25 msec)
Audible Ringing	1000 msec / (+/-200 msec)	3000 msec / (+/-2000 msec)
Audible Ringing (Tone 2)	2000 msec / (+/-200 msec)	4000 msec / (+/-2000 msec)

12. Resource Manager

In the described exemplary embodiment utilizing a multi-layer software architecture operating on a DSP platform, the DSP server includes networks VHDs (see FIG. 2). Each network VHD can be a complete self-contained software module for processing a single channel with a number of different telephony devices. Multiple channel capability can be achieved by adding network VHDs to the DSP server. The resource manager dynamically controls the creation and deletion of VHDs and services.

In the case of multi-channel communications using a number of network VHDs, the services invoked by the network VHDs and the associated PXDs are preferably optimized to minimize system resource requirements in terms of memory and/or computational complexity. This can be accomplished with the resource manager which reduces the complexity of certain algorithms in the network VHDs based on predetermined criteria. Although the resource management processor is described in the context of a signal processing system for packet voice exchange, those skilled in the art will appreciate that the techniques described for resource management processing are likewise suitable for various applications requiring processor complexity reductions. Accordingly, the described exemplary embodiment for resource management processing in a signal processing system is by way of example only and not by way of limitation.

The resource manager can be implemented to reduce complexity when the worst case system loading exceeds the peak system resources. The worst case system loading is simply the sum of the worst case (peak) loading of each service invoked by the network VHD and its associated PXDs.

1 The statistical nature of the processor resources required to process voice band telephony
signals is such that it is extremely unlikely that the worst case processor loading for each PXD
and /or service will occur simultaneously. Thus, a more robust (lower overall power
consumption and higher densities, i.e. more channels per DSP) signal processing system may be
5 realized if the average complexity of the various voice mode PXDs and associated services is
minimized. In the described exemplary embodiment, average system complexity is reduced and
system resources may be over subscribed (peak loading exceeds peak system resources) in the
short term wherein complexity reductions are invoked to reduce the peak loading placed on the
system.

10 The described exemplary resource manager should preferably manage the internal and
external program and data memory of the DSP. The transmission / signal processing of voice
is inherently dynamic, so that the system resources required for various stages of a conversation
are time varying. The resource manager should monitor DSP resource utilization and
dynamically allocate resources to numerous VHDs and PXDs to achieve a memory and
15 computationally (reduced MIPS) efficient system. For example, when the near end talker is
actively speaking, the voice encoder consumes significant resources, but the far end is probably
silent so that the echo canceller is probably not adapting and may not be executing the transversal
filter. When the far end is active, the near end is most likely inactive, which implies the echo
canceller is both canceling far end echo and adapting. However, when the far end is active the
near end is probably inactive, which implies that the VAD is probably detecting silence and the
20 voice encoder consumes minimal system resources. Thus, it is unlikely that the voice encoder
and echo canceller resource utilization peak simultaneously. Furthermore, if processor resources
are taxed, echo canceller adaptation may be disabled if the echo canceller is adequately adapted
or interleaved (adaptation enabled on alternating echo canceller blocks) to reduce the
computational burden placed on the processor.

25 Referring to FIG. 19, in the described exemplary embodiment, the resource manager 351
manages the resources of two network VHDs 62', 62" and their associated PXDs 60', 60".
Initially, the average complexity of the services running in each VHD and its associated PXD is
reported to the resource manager. The resource manager 351 sums the reported complexities to
determine whether the sum exceeds the system resources. If the sum of the average complexities
30 reported to the resource manager 351 are within the capability of the system resources, no
complexity reductions are invoked by the resource manager 351. Conversely, if the sum of the
average complexities of the services running in each VHD and its associated PXD overload the
system resources, then the resource manager can invoke a number of complexity reduction
methodologies. For example, the echo cancellers 70', 70" can be forced into the bypass mode
(see FIG. 7) and/or the echo canceller adaption can be reduced or disabled. In addition (or in the
35 alternative), complexity reductions in the voice encoders 82', 82" and voice decoders 96', 96" can
be invoked.

The described exemplary embodiment may reduce the complexity of certain voice mode services and associated PXDs so as to reduce the computational / memory requirements placed upon the system. Various modifications to the voice encoders may be included to reduce the load placed upon the system resources. For example, the complexity of a G.723.1 voice encoder may be reduced by disabling the post filter in accordance with the ITU-T G.723.1 standard which is incorporated herein by reference as if set forth in full. Also the voicing decision may be modified so as to be based on the open loop normalized pitch correlation computed at the open loop pitch lag L determined by the standard voice encoding algorithm. This entails a modification to the ITU-T G.723.1 C language routine `Estim_Pitch()`. If $d(n)$ is the input to the pitch estimation function, the normalized open loop pitch correlation at the open loop pitch lag L is:

$$X(L) = \frac{\sum_{n=0}^{N-1} (d(n)(d(n-L)))^2}{\left(\sum_{n=0}^{N-1} d(n) \right) \left(\sum_{n=0}^{N-1} d(n-L) \right)}$$

where N is equal to a duration of 2 subframes (or 120 samples).

Also, the ability to bypass the adaptive codebook based on a threshold computed from a combination of the open loop normalized pitch correlation and speech/residual energy may be included. In the standard encoder, the search through the adaptive codebook gain codebook begins at index zero and may be terminated before the entire codebook is searched (less than the total size of the adaptive codebook gain codebook which is either 85 or 170 entries) depending on the accumulation of potential error. A preferred complexity reduction truncates the adaptive codebook gain search procedure if the open loop normalized pitch correlation and speech/residual energy meets a certain by searching entries from:

- the upper bound (computed in the standard coder) less half the adaptive codebook size (or index zero, whichever is greater) for voiced speech; and

- from index zero up to half the size of the adaptive code gain codebook (85/2 or 170/2).

The adaptive codebook may also be completely bypassed under some conditions by setting the adaptive codebook gain index to zero, which selects an all zero adaptive codebook gain setting.

The fixed excitation in the standard encoder may have a periodic component. In the standard encoder, if the open loop pitch lag is less than the subframe length minus two, then a excitation search function (the function call `Find_Best()` in the ITU-T G.723.1 C language simulation) is invoked twice. To reduce system complexity, the fixed excitation search procedure may be modified (at 6.3 kb/s) such that the fixed excitation search function is invoked once per

1 invocation of the fixed excitation search procedure (routine Find_Fcbk()). If the open loop pitch
lag is less than the subframe length minus two then a periodic repetition is forced, otherwise there
is no periodic repetition (as per the standard encoder for that range of open loop pitch lags). In
the described complexity reduction modification, the decision on which manner to invoke it is
5 based on the open loop pitch lag and the voicing strength.

Similarly, the fixed excitation search procedure can be modified (at 5.3 kb/s) such that
a higher threshold is chosen for voice decisions. In the standard encoder, the voicing decision is
considered to be voiced if the open loop normalized pitch correlation is greater than 0.5 (variable
named "threshold" in the ITU-T G.723.1 is set to 0.5). In a modification to reduce the
10 complexity of this function, the threshold may be set to 0.75. This greatly reduces the
complexity of the excitation search procedure while avoiding substantial impairment to the voice
quality.

Similar modifications may be made to reduce the complexity of a G.729 Annex A voice
encoder. For example, the complexity of a G.729 Annex A voice decoder may be reduced by
15 disabling the post filter in accordance with the G.729 Annex A standard which is incorporated
herein by reference as if set out in full. Also, the complexity of a G.729 Annex A voice encoder
may be further reduced by including the ability to bypass the adaptive codebook or reduce the
complexity of the adaptive codebook search significantly. In the standard voice encoder, the
adaptive codebook searches over a range of lags based on the open loop pitch lag. The adaptive
20 codebook bypass simply chooses the minimum lag. The complexity of the adaptive codebook
search may be reduced by truncating the adaptive codebook search such that fractional pitch
periods are not considered within the search (not searching the non-integer lags). These
modifications are made to the ITU-T G.729 Annex A, C language routine Pitch_fr3_fast(). The
complexity of a G.729 Annex A voice encoder may be further reduced by substantially reducing
25 the complexity of the fixed excitation search. The search complexity may be reduced by
bypassing the depth first search 4, phase A: track 3 and 0 search and the depth first search 4,
phase B: track 1 and 2 search.

Each modification reduces the computational complexity but also minimally reduces the
resultant voice quality. However, since the voice encoders are externally managed by the
30 resource manager to minimize occasional system resource overloads, the voice encoder should
predominately operate with no complexity reductions. The preferred embedded software
embodiment should include the standard code as well as the modifications required to reduce the
system complexity. The resource manager should preferably minimize power consumption and
computational cycles by invoking complexity reductions which have substantially no impact on
voice quality. The different complexity reductions schemes should be selected dynamically
35 based on the processing requirements for the current frame (over all voice channels) and the
statistics of the voice signals on each channel (voice level, voicing, etc).

1 Although complexity reductions are rare, the appropriate PXDs and associated services
invoked in the network VHDs should preferably incorporate numerous functional features to
accommodate such complexity reductions. For example, the appropriate voice mode PXDs and
associated services should preferably include a main routine which executes the complexity
5 reductions described above with a variety of complexity levels. For example, various complexity
levels may be mandated by setting various complexity reduction flags. In addition, the resource
manager should accurately measure the resource requirements of PXDs and services with fixed
resource requirements (i.e. complexity is not controllable), to support the computation of peak
complexity and average complexity. Also, a function that returns the estimated complexity in
cycles according to the desired complexity reduction level should preferably be included.

10 The described exemplary embodiment preferably includes four complexity reduction
levels. In the first level, all complexity reductions are disabled so that the complexity of the
PXDs and services is not reduced.

15 The second level provides minimal or transparent complexity reductions (reductions
which should preferably have substantially no observable impact on performance under most
conditions). In the transparent mode the voice encoders (G.729, G.723.1) preferably use
voluntary reductions and the echo canceller is forced into the bypass mode and adaption is
toggled (i.e., adaptive is enabled for every other frame). Voluntary reductions for G.723.1 voice
encoders are preferably selected as follows. First, if the frame energy is less than -55 dBm0, then
20 the adaptive codebook is bypassed and the fixed excitation searches are reduced, as per above.
If the frame energy is less than -45 dBm0 but greater than -55 dBm0, then the adaptive codebook
is partially searched and the fixed excitation searches are reduced as per above. In addition, if
the open loop normalized pitch correlation is less than 0.305 then the adaptive codebook is
partially searched. Otherwise, no complexity reductions are done. Similarly, voluntary
25 reductions for the G.729 voice encoders preferably proceed as follows: first, if the frame energy
is less than -55 dBm0, then the adaptive codebook is bypassed and the fixed excitation search is
reduced per above. Next if the frame energy is less than -45 dBm0 but greater than -55 dBm0,
then the reduced complexity adaptive codebook is used and the excitation search complexity is
reduced. Otherwise, no complexity reduction is used.

30 The third level of complexity reductions provides minor complexity reductions
(reductions which may result in a slight degradation of voice quality or performance). For
example, in the third level the voice encoders preferably use voluntary reductions, "find_best"
reduction (G.723.1), fixed codebook threshold change (5.3 kbps G.723.1), open loop pitch search
reduction (G.723.1 only), and minimal adaptive codebook reduction (G.729 and G.723.1). In
35 addition, the echo canceller is forced into the bypass mode and adaption is toggled.

In the fourth level major complexity reductions occur, that is reductions which should
noticeably effect the performance quality. For example, in the fourth level of complexity
reductions the voice encoders use the same complexity reductions as those used for level three

1 reductions, as well as adding a bypass adaptive codebook reduction (G.729 and G.723.1). In
addition, the echo canceller is forced into the bypass mode and adaption is completely disabled.
The resource manager preferably limits the invocation of fourth level major reductions to extreme
circumstances, such as, for example when there is double talk on all active channels.

5 The described exemplary resource manager monitors system resource utilization. Under
normal system operating conditions, complexity reductions are not mandated on the echo
canceller or voice encoders. Voice/FAX and data traffic is packetized and transferred in packets.
The echo canceller removes echos, the DTMF detector detects the presence of keypad signals,
10 the VAD detects the presence of voice, and the voice encoders compress the voice traffic into
packets. However, when system resources are overtaxed and complexity reductions are required
there are at least two methods for controlling the voice encoder. In the first method, the
complexity level for the current frame is estimated from the information obtained from previous
voice frames and from the information gained from the echo canceller on the current voice frame.
The resource manager then mandates complexity reductions for the processing of frames in the
15 current frame interval in accordance with these estimations.

Alternatively, the voice encoders may be divided into a "front end" and a "back end".
The front end performs voice activity detection and open loop pitch detection (in the case of
G.723.1 and G.729 Annex A) on all channels operating on the DSP. Subsequent to the execution
of the front end function for all channels of a particular voice encoder, the system complexity
20 may be estimated based on the known information. Complexity reductions may then be
mandated to ensure that the current processing cycle can satisfy the processing requirements of
the voice encoders and decoders. This alternative method is preferred because the state of the
VAD is known whereas in the previously described method the state of the VAD is estimated.

25 In the alternate method, once the front end processing is complete so that the state of the
VAD and the voicing state for all channels is known, the system complexity may be estimated
based on the known statistics for the current frame. In the first method, the state of the VAD and
the voicing state may be estimated based on available known information. For example, the echo
canceller processes a voice encoder input signal to remove line echos prior to the activation of
the voice encoder. The echo canceller may estimate the state of the VAD based on the power
30 level of a reference signal and the voice encoder input signal so that the complexity level of all
controllable PXDs and services may be updated to determine the estimated complexity level of
each assuming no complexity reductions have been invoked. If the sum of all the various
complexity estimates is less than the complexity budget, no complexity reductions are required.
Otherwise, the complexity level of all system components are estimated assuming the invocation
of the transparent complexity reduction method to determine the estimated complexity resources
35 required for the current processing frame. If the sum of the complexity estimates with
transparent complexity reductions in place is less than the complexity budget, then the
transparent complexity reduction is used for that frame. In a similar manner, more and more

1 severe complexity reduction is considered until system complexity satisfies the prescribed budget.

5 The operating system should preferably allow processing to exceed the real-time constraint, i.e. maximum processing capability for the underlying DSP, in the short term. Thus data that should normally be processed within a given time frame or cycle may be buffered and processed in the next sequence. However, the overall complexity or processor loading must remain (on average) within the real-time constraint. This is a tradeoff between delay/jitter and channel density. Since packets may be delayed (due to processing overruns) overall end to end delay may increase slightly to account for the processing jitter.

10 Referring to FIG. 7, a preferred echo canceller has been modified to include an echo canceller bypass switch that invokes an echo suppressor in lieu of echo cancellation under certain system conditions so as to reduce processor loading. In addition, in the described exemplary embodiment the resource manager may instruct the adaptation logic 136 to disable filter adapter 134 so as to reduce processor loading under real-time constraints. The system will preferably limit adaptation on a fair and equitable basis when processing overruns occur. For example, if 15 four echo cancellers are adapting when a processing over run occurs, the resource manager may disable the adaption of echo cancellers one and two. If the processing over run continues, the resource manger should preferably enable adaption of echo cancellers one and two, and reduce system complexity by disabling the adaptation of echo cancellers three and four. This limitation should preferably be adjusted such that channels which are fully adapted have adaptation disabled first. In the described exemplary embodiment, the operating system should preferably control the subfunctions to limit peak system complexity. The subfunctions should be co-operative and include modifications to the echo canceller and the speech encoders.

25 B. The Fax Relay Mode

30 Fax relay mode provides signal processing of fax signals. As shown in FIG. 20, fax relay mode enables the transmission of fax signals over a packet based system such as VoIP, VoFR, FRF-11, VTOA, or any other proprietary network. The fax relay mode should also permit data signals to be carried over traditional media such as TDM. Network gateways 378a, 378b, 378c, the operating platform for the signal processing system in the described exemplary embodiment, support the exchange of fax signals between a packet based network 376 and various fax machines 380a, 380b, 380c. For the purposes of explanation, the first fax machine is a sending fax 380a. The sending fax 380a is connected to the sending network gateway 378a through a PSTN line 374. The sending network gateway 378a is connected to a packet based network 376. 35 Additional fax machines 380b, 380c are at the other end of the packet based network 376 and include receiving fax machines 380b, 380c and receiving network gateways 378b, 378c. The receiving network gateways 378b, 378b may provide a direct interface between their respective fax machines 380b, 380c and the packet based network 376.

1 The transfer of fax signals over packet based networks may be accomplished by at least
three alternative methods. In the first method, fax data signals are exchanged in real time.
Typically, the sending and receiving fax machines are spoofed to allow transmission delays plus
jitter of up to about 1.2 seconds. The second, store and forward mode, is a non real time method
5 of transferring fax data signals. Typically, the fax communication is transacted locally, stored
into memory and transmitted to the destination fax machine at a subsequent time. The third
mode is a combination of store and forward mode with minimal spoofing to provide an
approximate emulation of a typical fax connection.

10 In the fax relay mode, the network VHD invokes the packet fax data exchange. The
packet fax data exchange provides demodulation and re-modulation of fax data signals. This
approach results in considerable bandwidth savings since only the underlying unmodulated data
signals are transmitted across the packet based network. The packet fax data exchange also
provides compensation for network jitter with a jitter buffer similar to that invoked in the packet
voice exchange. Additionally, the packet fax data exchange compensates for lost data packets
15 with error correction processing. Spoofing may also be provided during various stages of the
procedure between the fax machines to keep the connection alive.

20 The packet fax data exchange is divided into two basic functional units, a demodulation
system and a re-modulation system. In the demodulation system, the network VHD couples fax
data signals from a circuit switched network, or a fax machine, to the packet based network. In
the re-modulation system, the network VHD couples fax data signals from the packet network
to the switched circuit network, or a fax machine directly.

25 During real time relay of fax data signals over a packet based network, the sending and
receiving fax machines are spoofed to accommodate network delays plus jitter. Typically, the
packet fax data exchange can accommodate a total delay of up to about 1.2 seconds. Preferably,
the packet fax data exchange supports error correction mode (ECM) relay functionality, although
a full ECM implementation is typically not required. In addition, the packet fax data exchange
should preferably preserve the typical call duration required for a fax session over a PSTN/ISDN
when exchanging fax data signals between two terminals.

30 The packet fax data exchange for the real time exchange of fax data signals between a
circuit switched network and a packet based network is shown schematically in FIG. 21. In this
exemplary embodiment, a connecting PXD (not shown) connecting the fax machine to the switch
board 32' is transparent, although those skilled in the art will appreciate that various signal
conditioning algorithms could be programmed into PXD such as echo cancellation and gain.

35 After the PXD (not shown), the incoming fax data signal 390a is coupled to the
demodulation system of the packet fax data exchange operating in the network VHD via the
switchboard 32'. The incoming fax data signal 390a is received and buffered in an ingress media
queue 390. A V.21 data pump 392 demodulates incoming T.30 message so that T.30 relay logic

1 394 can decode the received T.30 messages 394a. Local T.30 indications 394b are packetized
by a packetization engine 396 and if required, translated into T.38 packets via a T.38 shim 398
for transmission to a T.38 compliant remote network gateway (not shown) across the packet
based network. The V.21 data pump 392 is selectively enabled/disabled 394c by the T.30 relay
5 logic 394 in accordance with the reception/ transmission of the T.30 messages or fax data signals.
The V.21 data pump 392 is common to the demodulation and re-modulation system. The V.21
data pump 392 communicates T.30 messages such as for example called station tone (CED) and
calling station tone (CNG) to support fax setup between a local fax device (not shown) and a
remote fax device (not shown) via the remote network gateway.

10 The demodulation system further includes a receive fax data pump 400 which
demodulates the fax data signals during the data transfer phase. The receive fax data pump 400
supports the V.27ter standard for fax data signal transfer at 2400/4800 bps, the V.29 standard for
fax data signal transfer at 7200/9600 bps, as well as the V.17 standard for fax data signal transfer
at 7200/9600/12000/14400 bps. The V.34 fax standard, once approved, may also be supported.
15 The T.30 relay logic 394 enables / disables 394d the receive fax data pump 400 in accordance
with the reception of the fax data signals or the T.30 messages.

If error correction mode (ECM) is required, receive ECM relay logic 402 performs high
level data link control(HDLC)de-framing, including bit de-stuffing and preamble removal on
ECM frames contained in the data packets. The resulting fax data signals are then packetized by
20 the packetization engine 396 and communicated across the packet based network. The T.30 relay
logic 394 selectively enables / disables 394e the receive ECM relay logic 402 in accordance with
the error correction mode of operation.

25 In the re-modulation system, if required, incoming data packets are first translated from
a T.38 packet format to a protocol independent format by the T.38 packet shim 398. The data
packets are then de-packetized by a depacketizing engine 406. The data packets may contain
T.30 messages or fax data signals. The T.30 relay logic 394 reformats the remote T.30
indications 394f and forwards the resulting T.30 indications to the V.21 data pump 392. The
modulated output of the V.21 data pump 392 is forwarded to an egress media queue 408 for
transmission in either analog format or after suitable conversion, as 64 kbps PCM samples to the
30 local fax device over a circuit switched network, such as for example a PSTN line.

De-packetized fax data signals are transferred from the depacketizing engine 406 to a
jitter buffer 410. If error correction mode (ECM) is required, transmitting ECM relay logic 412
performs HDLC de-framing, including bit stuffing and preamble addition on ECM frames. The
transmitting ECM relay logic 412 forwards the fax data signals, (in the appropriate format) to a
35 transmit fax data pump 414 which modulates the fax data signals and outputs 8 KHz digital
samples to the egress media queue 408. The T.30 relay logic selectively enables/disables (394g)
the transmit ECM relay logic 412 in accordance with the error correction mode of operation.

1 The transmit fax data pump 414 supports the V.27ter standard for fax data signal transfer
at 2400/4800 bps, the V.29 standard for fax data signal transfer at 7200/9600 bps, as well as the
V.17 standard for fax data signal transfer at 7200/9600/12000/14400 bps. The T.30 relay logic
selectively enables/disables (394h) the transmit fax data pump 414 in accordance with the
transmission of the fax data signals or the T.30 message samples.

5 If the jitter buffer 410 underflows, a buffer low indication 410a is coupled to spoofing
logic 416. Upon receipt of a buffer low indication during the fax data signal transmission, the
spoofing logic 416 inserts "spoofed data" at the appropriate place in the fax data signals via the
transmit fax data pump 414 until the jitter buffer 410 is filled to a pre-determined level, at which
time the fax data signals are transferred out of the jitter buffer 410. Similarly, during the
transmission of the T.30 message indications, the spoofing logic 416 can insert "spoofed data"
at the appropriate place in the T.30 message samples via the V.21 data pump 392.

1. Data Rate Management

15 An exemplary embodiment of the packet fax data exchange complies with the T.38
recommendations for real-time Group 3 facsimile communication over packet based networks.
In accordance with the T.38 standard, the preferred system should therefore, provide packet fax
data exchange support at both the T.30 level (see ITU Recommendation T.30 - "Procedures for
Document Facsimile Transmission in the General Switched Telephone Network", 1988) and the
T4 level (see ITU Recommendation T.4 - "Standardization of Group 3 Facsimile Apparatus For
Document Transmission", 1998); the contents of each of these ITU recommendations being
incorporated herein by reference as if set forth in full. One function of the packet fax data
exchange is to relay the set up (capabilities) parameters in a timely fashion. Spoofing may be
needed at either or both the T.30 and T.4 levels to maintain the fax session while set up
parameters are negotiated at each of the network gateways and relayed in the presence of network
delays and jitter.

25 In accordance with the industry T.38 recommendations for real time Group 3
communication over packet based networks, the described exemplary embodiment relays all
information including; T.30 preamble indications (flags), T.30 message data, as well as T.30
image data between the network gateways. The T.30 relay logic 394 in the sending and receiving
network gateways then negotiate parameters as if connected via a PSTN line. The T.30 relay
logic 394 interfaces with the V.21 data pump 392 and the receive and transmit data pumps 400
and 414 as well as the packetization engine 396 and the depacketizing engine 406 to ensure that
the sending and the receiving fax machines 380(a) and 380(b) successfully and reliably
communicate. The T.30 relay logic 394 provides local spoofing, using command repeats (CRP),
and automatic repeat request (ARQ) mechanisms, incorporated into the T.30 protocol, to handle
delays associated with the packet based network. In addition, the T.30 relay logic 394 intercepts
control messages to ensure compatibility of the rate negotiation between the near end and far end

1 machines including HDLC processing, as well as lost packet recovery according to the T.30 ECM standard.

5 FIG. 22 demonstrates message flow over a packet based network between a sending fax machine 380a (see FIG. 20) and the receiving fax device 380b (see FIG. 20) in non-ECM mode. The PSTN fax call is divided into five phases: call establishment, control and capabilities exchange, page transfer, end of page and multi-page signaling and call release. In the call establishment phase, the sending fax machine dials the sending network gateway 378a (see FIG. 20) which forwards calling tone (CNG) (not shown) to the receiving network gateway 378b (see FIG. 20). The receiving network gateway responds by alerting the receiving fax machine. The receiving fax machine answers the call and sends called station (CED) tones. The CED tones are detected by the V.21 data pump 392 of the receiving network gateway which issues an event 420 indicating the receipt of CED which is then relayed to the sending network gateway. The sending network gateway forwards the CED tone 422 to the sending fax device. In addition, the V.21 data pump of the receiving network gateway invokes the packet fax data exchange.

15 In the control and capabilities exchange, the receiving network gateway transmits T.30 preamble (HDLC flags) 424 followed by called subscriber identification (CSI) 426 and digital identification signal (DIS) 428 message which contains the capabilities of the receiving fax device. The sending network gateway, forwards the HDLC flags, CSI and DIS to the sending fax device. Upon receipt of CSI and DIS, the sending fax device determines the conditions for the call by examining its own capabilities table relative to those of the receiving fax device. The sending fax device issues a command to the sending network gateway 430 to begin transmitting HDLC flags. Next, the sending fax device transmits subscriber identification (TSI) 432 and digital command signal (DCS) 434 messages, which define the conditions of the call to the sending network gateway. In response, the sending network gateway forwards V.21 HDLC sending subscriber identification / frame check sequences and digital command signal / frame check sequences to the receiving fax device via the receiving network gateway. Next the sending fax device transmits training check (TCF) fields 436 to verify the training and ensure that the channel is suitable for transmission at the accepted data rate.

30 The TCF 436 may be managed by one of two methods. The first method, referred to as the data rate management method one in the T.38 standard, the receiving network gateway locally generate TCF. Confirmation to receive (CFR) is returned to the sending fax device 380(a), when the sending network gateway receives a confirmation to receive (CFR) 438 from the receiving fax machine via the receiving network gateway, and the TCF training 436 from the sending fax machine is received successfully. In the event that the receiving fax machine receives a CFR and the TCF training 436 from the sending fax machine subsequently fails, then DCS 434 from the sending fax machine is again relayed to the receiving fax machine. The TCF training 436 is repeated until an appropriate rate is established which provides successful TCF training 436 at both ends of the network.

1 In a second method to synchronize the data rate, referred to as the data rate management
method two in the T.38 standard, the TCF data sequence received by the sending network
gateway is forwarded from the sending fax machine to the receiving fax machine via the
receiving network gateway. The sending and receiving fax machines then perform speed
5 selection as if connected via a regular PSTN.

Upon receipt of confirmation to receive (CFR) 440 which indicates that all capabilities
and the modulation speed have been confirmed, the sending fax machine enters the page transfer
phase, and transmits image data 444 along with its training preamble 442. The sending network
gateway receives the image data and forwards the image data 444 to the receiving network
10 gateway. The receiving network gateway then sends its own training preamble 446 followed by
the image data 448 to the receiving fax machine.

In the end of page and multi-page signaling phase, after the page has been successfully
transmitted, the sending fax device sends an end of procedures (EOP) 450 message if the fax call
15 is complete and all pages have been transmitted. If only one of multiple pages has been
successfully transmitted, the sending fax device transmits a multi-page signal (MPS). The
receiving fax device responds with message confirmation (MCF) 452 to indicate the message has
been successfully received and that the receiving fax device is ready to receive additional pages.
The release phase is the final phase of the call, where at the end of the final page, the receiving
fax machine sends a message confirmation (MCF) 452, which prompts the sending fax machine
20 to transmit a disconnect (DCN) signal 454. The call is then terminated at both ends of the
network.

ECM fax relay message flow is similar to that described above. All preambles, messages
and page transfers (phase C) HDLC data are relayed through the packet based network. Phase
25 C HDLC data is de-stuffed and, along with the preamble and frame checking sequences (FCS),
removed before being relayed so that only fax image data itself is relayed over the packet based
network. The receiving network gateway performs bit stuffing and reinserts the preamble and
FCS.

30 2. Spoofing Techniques

Spoofing refers to the process by which a facsimile transmission is maintained in the
presence of data packet under-run due to severe network jitter or delay. An exemplary
embodiment of the packet fax data exchange complies with the T.38 recommendations for real-
time Group 3 facsimile communication over packet based networks. In accordance with the T.38
35 recommendations, a local and remote T.30 fax device communicate across a packet based
network via signal processing systems, which for the purposes of explanation are operating in
network gateways. In operation, each fax device establishes a facsimile connection with its
respective network gateway in accordance with the ITU-T.30 standards and the signal processing
systems operating in the network gateways relay data signals across a packet based network.

1 In accordance with the T.30 protocol, there are certain time constraints on the
handshaking and image data transmission for the facsimile connection between the T.30 fax
device and its respective network gateway. The problem that arises is that the T.30 facsimile
protocol is not designed to accommodate the significant jitter and packet delay that is common
5 to communications across packet based networks. To prevent termination of the fax connection
due to severe network jitter or delay, it is, therefore, desirable to ensure that both T.30 fax devices
can be spoofed during periods of data packet under-run. FIG. 23 demonstrates fax
communication 466 under the T.30 protocol, wherein a handshake negotiator 468, typically a low
speed modem such as V.21, performs handshake negotiation and fax image data is communicated
10 via a high speed data pump 470 such as V.27, V.29 or V.17. In addition, fax image data can be
transmitted in an error correction mode (ECM) 472 or non error correction mode (non-ECM)
474, each of which uses a different data format.

Therefore, in the described exemplary embodiment, the particular spoofing technique
utilized is a function of the transmission format. In the described exemplary embodiment, HDLC
15 preamble 476 is used to spoof the T.30 fax devices during V.21 handshaking and during
transmission of fax image data in the error correction mode. However, zero-bit filling 478 is
used to spoof the T.30 fax devices during fax image data transfer in the non error correction
mode. Although fax relay spoofing is described in the context of a signal processing system with
the packet data fax exchange invoked, those skilled in the art will appreciate that the described
exemplary fax relay spoofing method is likewise suitable for various other telephony and
20 telecommunications application. Accordingly, the described exemplary embodiment of fax relay
spoofing in a signal processing system is by way of example only and not by way of limitation.

a. V.21 HDLC Preamble Spoofing

25 The T.30 relay logic 394 packages each message or command into a HDLC frame which
includes preamble flags. An HDLC frame structure is utilized for all binary-coded V.21
facsimile control procedures. The basic HDLC structure consists of a number of frames, each
of which is subdivided into a number of fields. The HDLC frame structure provides for frame
labeling and error checking. When a new facsimile transmission is initiated, HDLC preamble
30 in the form of synchronization sequences are transmitted prior to the binary coded information.
The HDLC preamble is V.21 modulated bit streams of "01111110 (0x7e)".

35 In the described exemplary embodiment, spoofing techniques are utilized at the T.30 and
T.4 levels to manage extended network delays and jitter. Turning back to FIG. 21, the T.30 relay
logic 394 waits for a response to any message or command transmitted across the packet based
network before continuing to the next state or phase. In accordance with an exemplary spoofing
technique, the sending and receiving network gateways 378a, 378b (See FIG. 20) spoof their
respective fax machines 380a, 380b by locally transmitting HDLC preamble flags if a response
to a transmitted message is not received from the packet based network within approximately

1 1.5-2.0 seconds. The maximum length of the preamble is limited to about four seconds. If a
response from the packet based network arrives before the spoofing time out, each network
gateway should preferably transmit a response message to its respective fax machine following
the preamble flags. Otherwise, if the network response to a transmitted message is not received
5 prior to the spoofing time out (in the range of about 5.5-6.0 seconds), the response is assumed
to be lost. In this case, when the network gateway times out and terminates preamble spoofing,
the local fax device transmits the message command again. Each network gateway repeats the
spoofing technique until a successful handshake is completed or its respective fax machine
disconnects.

10 b. ECM HDLC Preamble Spoofing

The packet fax data exchange utilizes an HDLC frame structure for ECM high-speed data
transmission. Preferably, the frame image data is divided by one or more HDLC preamble flags.
If the network under-runs due to jitter or packet delay, the network gateways spoof their
15 respective fax devices at the T.4 level by adding extra HDLC flags between frames. This
spoofing technique increases the sending time to compensate for packet under-run due to
network jitter and delay. Returning to FIG. 21 if the jitter buffer 410 underflows, a buffer low
indication 410a is coupled to the spoofing logic 416. Upon receipt of a buffer low indication
during the fax data signal transmission, the spoofing logic 416 inserts HDLC preamble flags at
the frame boundary via the transmit fax data pump 414. When the jitter buffer 410 is filled to
20 a pre-determined level, the fax image data is transferred out of the jitter buffer 410.

In the described exemplary embodiment, the jitter buffer 410 must be sized to store at
least one HDLC frame so that a frame boundary may be located. The length of the largest T.4
ECM HDLC frame is 260 octets or 130 16-bit words. Spoofing is preferably activated when the
25 number of packets stored in the jitter buffer 410 drops to a predetermined threshold level. When
spoofing is required, the spoofing logic 416 adds HDLC flags at the frame boundary as a
complete frame is being reassembled and forwarded to the transmit fax data pump 414. This
continues until the number of data packets in the jitter buffer 410 exceeds the threshold level.
The maximum time the network gateways will spoof their respective local fax devices can vary
but can generally be about ten seconds.

30 c. Non-ECM Spoofing with Zero Bit Filling

T.4 spoofing handles delay impairments during page transfer or C phase of a fax call. For
those systems that do not utilize ECM, phase C signals comprise a series of coded image data
followed by fill bits and end-of-line (EOL) sequences. Typically, fill bits are zeros inserted
35 between the fax data signals and the EOL sequences, "000000000001". Fill bits ensure that a fax
machine has time to perform the various mechanical overhead functions associated with any line
it receives. Fill bits can also be utilized to spoof the jitter buffer to ensure compliance with the
minimum transmission time of the total coded scan line established in the pre-message V.21

control procedure. The number of the bits of coded image contained in the data signals associated with the scan line and transmission speed limit the number of fill bits that can be added to the data signals. Preferably, the maximum transmission of any coded scan line is limited to less than about 5 sec. Thus, if the coded image for a given scan line contains 1000 bits and the transmission rate is 2400 bps, then the maximum duration of fill time is $(5 - (1000 + 12)/2400) = 4.57$ sec.

Generally, the packet fax data exchange utilizes spoofing if the network jitter delay exceeds the delay capability of the jitter buffer 410. In accordance with the EOL spoofing method, fill bits can only be inserted immediately before an EOL sequence, so that the jitter buffer 410 should preferably store at least one EOL sequence. Thus the jitter buffer 410 should preferably be sized to hold at least one entire scan line of data to ensure the presence of at least one EOL sequence within the jitter buffer 410. Thus, depending upon transmission rate, the size of the jitter buffer 410 can become prohibitively large. The table below summarizes the desired jitter buffer data space to perform EOL spoofing for various scan line lengths. The table assumes that each pixel is represented by a single bit. The values represent an approximate upper limit on the required data space, but not the absolute upper limit, because in theory at least, the longest scan line can consist of alternating black and white pixels which would require an average of 4.5 bits to represent each pixel rather than the one to one ratio summarized in the table.

Scan Line Length	Number of words	sec to print out at 2400	sec to print out at 4800	sec to print out at 9600	sec to print out at 14400
1728	108	0.72	0.36	0.18	0.12
2048	128	0.853	0.427	0.213	0.14
2432	152	1.01	0.507	0.253	0.17
3456	216	1.44	0.72	0.36	0.24
4096	256	2	0.853	0.43	0.28
4864	304	2.375	1.013	0.51	0.34

To ensure the jitter buffer 410 stores an EOL sequence, the spoofing logic 416 should be activated when the number of data packets stored in the jitter buffer 410 drops to a threshold level. Typically, a threshold value of about 200 msec is used to support the most commonly used fax setting, namely a fax speed of 9600 bps and scan line length of 1728. An alternate spoofing method should be used if an EOL sequence is not contained within the jitter buffer 410, otherwise the call will have to be terminated. An alternate spoofing method uses zero run length code words. This method requires real time image data decoding so that the word boundary is known. Advantageously, this alternate method reduces the required size of the jitter buffer 410.

1 Simply increasing the storage capacity of the jitter buffer 410 can minimize the need for
spoofing. However, overall network delay increases when the size of the jitter buffer 410 is
increased. Increased network delay may complicate the T.30 negotiation at the end of page or
end of document, because of susceptibility to time out. Such a situation arises when the sending
5 fax machine completes the transmission of high speed data, and switches to an HDLC phase and
sends the first V.21 packet in the end of page / multi-page signaling phase, (i.e. phase D). The
sending fax machine must be kept alive until the response to the V.21 data packet is received.
The receiving fax device requires more time to flush a large jitter buffer and then respond, hence
complicating the T.30 negotiation.

10 In addition, the length of time a fax machine can be spoofed is limited, so that the jitter
buffer 410 can not be arbitrarily large. A pipeline store and forward relay is a combination of
store and forward and spoofing techniques to approximate the performance of a typical Group
3 fax connection when the network delay is large (on the order of seconds or more). One
approach is to store and forward a single page at a time. However, this approach requires a
15 significant amount of memory (10 Kwords or more). One approach to reduce the amount of
memory required entails discarding scan lines on the sending network gateway and performing
line repetition on the receiving network gateway so as to maintain image aspect ratio and quality.
Alternatively, a partial page can be stored and forwarded thereby reducing the required amount
of memory.

20 The sending and receiving fax machines will have some minimal differences in clock
frequency. ITU standards recommends a data pump data rate of ± 100 ppm, so that the clock
frequencies between the receiving and sending fax machines could differ by up to 200 ppm.
Therefore, the data rate at the receiving network gateway (jitter buffer 410) can build up or
deplete at a rate of 1 word for every 5000 words received. Typically a fax page is less than 1000
25 words so that end to end clock synchronization is not required.

C. Data Relay Mode

30 Data relay mode provides full duplex signal processing of data signals. As shown in FIG.
24, data relay mode enables the transmission of data signals over a packet based system such as
VoIP, VoFR, FRF-11, VTOA, or any other proprietary network. The data relay mode should
also permit data signals to be carried over traditional media such as TDM. Network gateways
496a, 496b, 496c, support the exchange of data signals between a packet based network 494 and
various data modems 492a, 492b, 492c. For the purposes of explanation, the first modem is
referred to as a call modem 492a. The call modem 492a is connected to the call network gateway
496a through a PSTN line. The call network gateway 496a is connected to a packet based
35 network 494. Additional modems 492b, 492c are at the other end of the packet based network
494 and include answer modems 492b, 492c and answer network gateways 496b, 496c. The
answer network gateways 496b, 496c provide a direct interface between their respective modems
492b, 492c and the packet based network 494.

1 In data relay mode, a local modem connection is established on each end of the packet
based network 494. That is, the call modem 492a and the call network gateway 496a establish
a local modem connection, as does the destination answer modem 492b and its respective answer
network gateway 496b. Next, data signals are relayed across the packet based network 494. The
5 call network gateway 496a demodulates the data signal and formats the demodulated data signal
for the particular packet based network 494. The answer network gateway 496b compensates
for network impairments and remodulates the encoded data in a format suitable for the
destination answer modem 492b. This approach results in considerable bandwidth savings since
only the underlying demodulated data signals are transmitted across the packet based network.

10 In the data relay mode, the packet data modem exchange provides demodulation and
modulation of data signals. With full duplex capability, both modulation and demodulation of
data signals can be performed simultaneously. The packet data modem exchange also provides
compensation for network jitter with a jitter buffer similar to that invoked in the packet voice
exchange. Additionally, the packet data modem exchange compensates for system clock jitter
15 between modems with a dynamic phase adjustment and resampling mechanism. Spoofing may
also be provided during various stages of the call negotiation procedure between the modems to
keep the connection alive.

The packet data modem exchange invoked by the network VHD in the data relay mode
is shown schematically in FIG. 25. In the described exemplary embodiment, a connecting PXD
20 (not shown) connecting a modem to the switch board 32' is transparent, although those skilled
in the art will appreciate that various signal conditioning algorithms could be programmed into
the PXD such as filtering, echo cancellation and gain.

After the PXD, the data signals are coupled to the network VHD via the switchboard 32'.
25 The packet data modem exchange provides two way communication between a circuit switched
network and packet based network with two basic functional units, a demodulation system and
a remodulation system. In the demodulation system, the network VHD exchanges data signals
from a circuit switched network, or a telephony device directly, to a packet based network. In
the remodulation system, the network VHD exchanges data signals from the packet based
network to the PSTN line, or the telephony device.

30 In the demodulation system, the data signals are received and buffered in an ingress
media queue 500. A data pump receiver 504 demodulates the data signals from the ingress media
queue 500. The data pump receiver 504 supports the V.22bis standard for the demodulation of
data signals at 1200/2400 bps; the V.32bis standard for the demodulation of data signals at
4800/7200/9600/12000/14400 bps, as well as the V.34 standard for the demodulation of data
35 signals up to 33600 bps. Moreover, the V.90 standard may also be supported. The demodulated
data signals are then packetized by the packetization engine 506 and transmitted across the packet
based network.

1 In the remodulation system, packets of data signals from the packet based network are first depacketized by a depacketizing engine 508 and stored in a jitter buffer 510. A data pump transmitter 512 modulates the buffered data signals with a voiceband carrier. The modulated data signals are in turn stored in the egress media queue 514 before being output to the PXD (not shown) via the switchboard 32'. The data pump transmitter 512 supports the V.22bis standard for the transfer of data signals at 1200/2400 bps; the V.32bis standard for the transfer of data signals at 4800/7200/9600/12000/14400 bps, as well as the V.34 standard for the transfer of data signal up to 33600 bps. Moreover, the V.90 standard may also be supported.

10 During jitter buffer underflow, the jitter buffer 510 sends a buffer low indication 510a to spoofing logic 516. When the spoofing logic 516 receives the buffer low signal indicating that the jitter buffer 510 is operating below a predetermined threshold level, it inserts spoofed data at the appropriate place in the data signal via the data pump transmitter 512. Spoofing continues until the jitter buffer 510 is filled to the predetermined threshold level, at which time data signals are again transferred from the jitter buffer 510 to the data pump transmitter 512.

15 End to end clock logic 518 also monitors the state of the jitter buffer 510. The clock logic 518 controls the data transmission rate of the data pump transmitter 512 in correspondence to the state of the jitter buffer 510. When the jitter buffer 510 is below a predetermined threshold level, the clock logic 518 reduces the transmission rate of the data pump transmitter 512. Likewise, when the jitter buffer 510 is above a predetermined threshold level, the clock logic 518 increases the transmission rate of the data pump transmitter 512.

20 Before the transmission of data signals across the packet based network, the connection between the two modems must first be negotiated through a handshaking sequence. This entails a two-step process. First, a call negotiator 502 determines the type of modem (i.e., V.22, V.32bis, V.34, V.90, etc.) connected to each end of the packet based network. Second, a rate negotiator 520 negotiates the data signal transmission rate between the two modems.

25 The call negotiator 502 determines the type of modem connected locally, as well as the type of modem connected remotely via the packet based network. The call negotiator 502 utilizes V.25 automatic answering procedures and V.8 auto-baud software to automatically detect modem capability. The call negotiator 502 receives protocol indication signals 502a (ANSam and V.8 menus) from the ingress media queue 500, as well as AA, AC and other message indications 502b from the local modem via a data pump state machine 522, to determine the type of modem in use locally. The call negotiator 502 relays the ANSam answer tones and other indications 502e from the data pump state machine 522 to the remote modem via a packetization engine 506. The call negotiator also receives ANSam, AA, AC and other indications 502c from a remote modem (not shown) located on the opposite end of the packet based network via a depacketizing engine 508. The call negotiator 502 relays ANSam answer tones and other indications 502d to a local modem (not shown) via an egress media queue 514 of the modulation system. With the ANSam, AA, AC and other indications from the local and remote modems, the

1 call negotiator 502 can then negotiate a common standard (i.e., V.22, V.32bis, V.34, V.90, etc.)
in which the data pumps must communicate with the local modem and the remote modems.

5 The packet data modem exchange preferably utilizes indication packets as a means for
communicating answer tones, AA, AC and other indication signals across the packet based
network. However, the packet data modem exchange supports data pumps such as V.22bis and
V.32bis which do not include a well defined error recovery mechanism, so that the modem
connection may be terminated whenever indication packets are lost. Therefore, either the packet
data modem exchange or the application layer should ensure proper delivery of indication packets
when operating in a network environment that does not guarantee packet delivery.

10 The packet data modem exchange can ensure delivery of the indication packets by
periodically retransmitting the indication packet until some expected packets are received. For
example, in V.32bis relay, the call negotiator operating under the packet data modem exchange
on the answer network gateway periodically retransmits ANSam answer tones from the answer
modem to the call modem, until the calling modem connects to the line and transmits carrier
state AA.

15 Alternatively, the packetization engine can embed the indication information directly into
the packet header. In this approach, an alternate packet format is utilized to include the
indication information. During modem handshaking, indication packets transmitted across the
packet based network include the indication information, so that the system does not rely on the
successful transmission of individual indication packets. Rather, if a given packet is lost, the
next arriving packet contains the indication information in the packet header. Both methods
increase the traffic across the network. However, it is preferable to periodically retransmit the
indication packets because it has less of a detrimental impact on network traffic.

25 A rate negotiator 520 synchronizes the connection rates at the network gateways 496a,
496b, 496c (see FIG. 24). The rate negotiator receives rate control codes 520a from the local
modem via the data pump state machine 522 and rate control codes 520b from the remote modem
via the depacketizing engine 508. The rate negotiator 520 also forwards the remote rate control
codes 520a received from the remote modem to the local modem via commands sent to the data
pump state machine 522. The rate negotiator 520 forwards the local rate control codes 520c
received from the local modem to the remote modem via the packetization engine 506. Based on
the exchanged rate codes the rate negotiator 520 establishes a common data rate between the
calling and answering modems. During the data rate exchange procedure, the jitter buffer 510
should be disabled by the rate negotiator 520 to prevent data transmission between the call and
answer modems until the data rates are successfully negotiated.

35 Similarly error control (V.42) and data compression (V.42bis) modes should be
synchronized at each end of the packet based network. Error control logic 524 receives local
error control messages 524a from the data pump receiver 504 and forwards those V.14/V.42

1 negotiation messages 524c to the remote modem via the packetization engine 506. In addition,
error control logic 524 receives remote V.14/V.42 indications 524b from the depacketizing
engine 508 and forwards those V.14/V.42 indications 524d to the local modem. With the
V.14/V.42 indications from the local and remote modems, the error control logic 524 can
5 negotiate a common standard to ensure that the network gateways utilize a common error
protocol. In addition, error control logic 524, communicates the negotiated error control protocol
524(e) to the spoofing logic 516 to ensure data mode spoofing is in accordance with the
negotiated error control mode.

10 V.42 is a standard error correction technique using advanced cyclical redundancy checks
and the principle of automatic repeat requests (ARQ). In accordance with the V.42 standard,
transmitted data signals are grouped into blocks and cyclical redundancy calculations add error
checking words to the transmitted data signal stream. The receiving modem calculates new error
check information for the data signal block and compares the calculated information to the
15 received error check information. If the codes match, the received data signals are valid and
another transfer takes place. If the codes do not match, a transmission error has occurred and the
receiving modem requests a repeat of the last data block. This repeat cycle continues until the
entire data block has been received without error.

Various voiceband data modem standards exist for error correction and data compression.
V.42bis and MNP5 are examples of data compression standards. The handshaking sequence for
20 every modem standard is different so that the packet data modem exchange should support
numerous data transmission standards as well as numerous error correction and data compression
techniques.

1. End to End Clock Logic

25 Slight differences in the clock frequency of the call modem and the answer modem are
expected, since the baud rate tolerance for a typical modem data pump is ± 100 ppm. This
tolerance corresponds to a relatively low depletion or build up rate of 1 in 5000 words. However,
the length of a modem session can be very long, so that uncorrected difference in clock frequency
may result in jitter buffer underflow or overflow.

30 In the described exemplary embodiment, the clock logic synchronizes the transmit clock
of the data pump transmitter 512 to the average rate at which data packets arrive at the jitter
buffer 510. The data pump transmitter 512 packages the data signals from the jitter buffer 510
in frames of data signals for demodulation and transmission to the egress media queue 514. At
the beginning of each frame of data signals, the data pump transmitter 512 examines the egress
35 media queue 514 to determine the remaining buffer space, and in accordance therewith, the data
pump transmitter 512 modulates that number of digital data samples required to produce a total
of slightly more or slightly less than 80 samples per frame, assuming that the data pump
transmitter 512 is invoked once every 10 msec. The data pump transmitter 512 gradually adjusts

1 the number of samples per frame to allow the receiving modem to adjust to the timing change. Typically, the data pump transmitter 512 uses an adjustment rate of about one ppm per frame. The maximum adjustment should be less than about 200 ppm.

5 In the described exemplary embodiment, end to end clock logic 518 monitors the space available within the jitter buffer 510 and utilizes water marks to determine whether the data rate of the data pump transmitter 512 should be adjusted. Network jitter may cause timing adjustments to be made. However, this should not adversely affect the data pump receiver of the answering modem as these timing adjustments are made very gradually.

10 2. Modem Connection Handshaking Sequence.

a. Call Negotiation.

15 A single industry standard for the transmission of modem data over a packet based network does not exist. However, numerous common standards exist for transmission of modem data at various data rates over the PSTN. For example, V.22 is a common standard used to define operation of 1200 bps modems. Data rates as high as 2400 bps can be implemented with the V.22bis standard (the suffix "bis" indicates that the standard is an adaptation of an existing standard). The V.22bis standard groups data signals into four bit words which are transmitted at 600 baud. The V.32 standard supports full duplex, data rates of up to 9600 bps over the PSTN. 20 A V.32 modem groups data signals into four bit words and transmits at 2400 baud. The V.32bis standard supports duplex modems operating at data rates up to 14,400 bps on the PSTN. In addition, the V.34 standard supports data rates up to 33,600 bps on the public switched telephone network. In the described exemplary embodiment, these standards can be used for data signal transmission over the packet based network with a call negotiator that supports each standard.

25 b. Rate Negotiation.

30 Rate negotiation refers to the process by which two telephony devices are connected at the same data rate prior to data transmission. In the context of a modem connection in accordance with an exemplary embodiment of the present invention, each modem is coupled to a signal processing system, which for the purposes of explanation is operating in a network gateway, either directly or through a PSTN line. In operation, each modem establishes a modem connection with its respective network gateway, at which point, the modems begin relaying data signals across a packet based network. The problem that arises is that each modem may negotiate a different data rate with its respective network gateway, depending on the line 35 conditions and user settings. In this instance, the data signals transmitted from one of the modems will enter the packet based network faster than it can be extracted at the other end by the other modem. The resulting overflow of data signals may result in a lost connection between the two modems. To prevent data signal overflow, it is, therefore, desirable to ensure that both

1 modems negotiate to the same data rate. A rate negotiator can be used for this purpose.
Although the the rate negotiator is described in the context of a signal processing system with
the packet data modem exchange invoked, those skilled in the art will appreciate that the rate
negotiator is likewise suitable for various other telephony and telecommunications application.
5 Accordingly, the described exemplary embodiment of the rate negotiator in a signal processing
system is by way of example only and not by way of limitation.

In an exemplary embodiment, data rate negotiation is achieved through a data rate
negotiation procedure, wherein a call modem independently negotiates a data rate with a call
network gateway, and an answer modem independently negotiates a data rate with an answer
10 network gateway. The calling and answer network gateways, each having a signal processing
system running a packet exchange, then exchange data packets containing information on the
independently negotiated data rates. If the independently negotiated data rates are the same, then
each rate negotiator will enable its respective network gateway and data transmission between
the call and answer modems will commence. Conversely, if the independently negotiated data
15 rates are different, the rate negotiator will renegotiate the data rate by adopting the lowest of the
two data rates. The call and answer modems will then undergo retraining or rate renegotiation
procedures by their respective network gateways to establish a new connection at the renegotiated
data rate. The advantage of this approach is that the data rate negotiation procedure takes
advantage of existing modem functionality, namely, the retraining and rate renegotiation
mechanism, and puts it to alternative usage. Moreover, by retraining both the call and answer
20 modem (one modem will already be set to the renegotiated rate) both modems are automatically
prevented from sending data.

Alternatively, the calling and answer modems can directly negotiate the data rate. This
method is not preferred for modems with time constrained handshaking sequences such as, for
25 example, modems operating in accordance with the V.22bis or the V.32bis standards. The round
trip delay accommodated by these standards could cause the modem connection to be lost due
to timeout. Instead, retrain or rate renegotiation should be used for data signals transferred in
accordance with the V.22bis and V.32bis standards, whereas direct negotiation of the data rate
by the local and remote modems can be used for data exchange in accordance with the V.34 and
V.90 (a digital modem and analog modem pair for use on PSTN lines at data rates up to 56,000
30 bps downstream and 33,600 upstream) standards.

c. Exemplary Handshaking Sequences.

(V.22 Handshaking Sequence)

35 The call negotiator on the answer network gateway, differentiates between modem types
and relays the ANSam answer tone. The answer modem transmits unscrambled binary ones
signal (USB1) indications to the answer mode gateway. The answer network gateway forwards
USB1 signal indications to the call network gateway. The call negotiator in the call network

1 gateway assumes operation in accordance with the V.22bis standard as a result of the USB1
signal indication and terminates the call negotiator. The packet data modem exchange, in the
answer network gateway then invokes operation in accordance with the V.22bis standard after
an answer tone timeout period and terminates its call negotiator.

5 V.22bis handshaking does not utilize rate messages or signaling to indicate the selected
bit rate as with most high data rate pumps. Rather, the inclusion of a fixed duration signal (S1)
indicates that 2400 bps operation is to be used. The absence of the S1 signal indicates that 1200
bps should be selected. The duration of the S1 signal is typically about 100 msec, making it
10 likely that the call modem will perform rate determination (assuming that it selects 2400 bps)
before rate indication from the answer modem arrives. Therefore, the rate negotiator in the call
network gateway should select 2400 bps operation and proceed with the handshaking procedure.
If the answer modem is limited to a 1200 bps connection, rate renegotiation is typically used to
change the operational data rate of the call modem to 1200 bps. Alternatively, if the call modem
selects 1200 bps, rate renegotiation would not be required.

15 (V.32bis Handshaking Sequence)

V32bis handshaking utilizes rate signals (messages) to specify the bit rate. A relay
sequence in accordance with the V.32bis standard is shown in FIG. 26 and begins with the call
negotiator in the answer network gateway relaying ANSam 530 answer tone from the answer
20 modem to the call modem. After receiving the answer tone for a period of at least one second,
the call modem connects to the line and repetitively transmits carrier state A 532. When the call
network gateway detects the repeated transmission of carrier state A ("AA"), the call network
gateway relays this information 534 to the answer network gateway. In response the answer
network gateway forwards the AA indication to the answer modem and invokes operation in
25 accordance with the V.32bis standard. The answer modem then transmits alternating carrier
states A and C 536 to the answer network gateway. If the answer network gateway receives AC
from the answer modem, the answer network gateway relays AC 538 to the call network
gateway, thereby establishing operation in accordance with the V.32bis standard, allowing call
negotiator in the call network gateway to be terminated. Next, data rate alignment is achieved
by either of two methods.

30 In the first method for data rate alignment of a V.32bis relay connection, the call modem
and the answer modem independently negotiate a data rate with their respective network
gateways at each end of the network 540 and 542. Next, each network gateway forwards a
connection data rate indication 544 and 546 to the other network gateway. Each network
gateway compares the far end data rate to its own data rate. The preferred rate is the minimum
35 of the two rates. Rate renegotiation 548 and 550 is invoked if the connection rate of either
network gateway to its respective modem differs from the preferred rate.

In the second method, rate signals R1, R2 and R3, are relayed to achieve data rate

1 negotiation. FIG. 27 shows a relay sequence in accordance with the V.32bis standard for this
alternate method of rate negotiation. The call negotiator relays the answer tone (ANSam) 552
from the answer modem to the call modem. When the call modem detects answer tone, it
repetitively transmits carrier state A 554 to the call network gateway. The call network gateway
5 relays this information (AA) 556 to the answer network gateway. The answer network gateway
sends the AA 558 to the answer modem and initiates normal range tone exchange with the
answer modem. The answer network gateway then forwards AC 560 to call network gateway
which in turn relays this information 562 to the call modem to initiate normal range tone
exchange between the call network gateway and the call modem.

10 The answer modem sends its first training sequence 564 followed by R1 (the data rates
currently available in the answer modem) to the rate negotiator in the answer network gateway.
When the answer network gateway receives an R1 indication, it forwards R1 566 to the call
network gateway. The answer network gateway then repetitively sends training sequences to the
answer modem. The call network gateway forwards the R1 indication 570 of the answer modem
15 to the call modem. The call modem sends training sequences to the call network gateway 572.
The call network gateway determines the data rate capability of the call modem, and forwards
the data rate capabilities of the call modem to the answer network gateway in a data rate signal
format. The call modem also sends an R2 indication 568 (data rate capability of the call modem,
preferably excluding rates not included in the previously received R1 signal, i.e. not supported
by the answer modem) to the call network gateway which forwards it to the answer network
20 gateway. The call network gateway then repetitively sends training sequences to the call modem
until receiving an R3 signal 574 from the answer modem via the answer network gateway.

The answer network gateway performs a logical AND operation on the R1 signal from
the answer modem (data rate capability of the answer modem), the R2 signal from the call
25 modem (data rate capability of the call modem, excluding rates not supported by the answer
modem) and the training sequences of the call network gateway (data rate capability of the call
modem) to create a second rate signal R2 576, which is forwarded to the answer modem. The
answer modem sends its second training sequence followed an R3 signal, which indicates the
data rate to be used by both modems. The answer network gateway relays R3 574 to the call
network gateway which forwards it to the call modem and begins operating at the R3 specified
30 bit rate. However, this method of rate synchronization is not preferred for V.32bis due to time
constrained handshaking.

(V.34 Handshaking Sequence)

35 Data transmission in accordance with the V.34 standard utilizes a modulation parameter
(MP) sequence to exchange information pertaining to data rate capability. The MP sequences
can be exchanged end to end to achieve data rate synchronization. Initially, the call negotiator
in the answer network gateway relays the answer tone (ANSam) from the answer modem to the
call modem. When the call modem receives answer tone, it generates a CM indication and

1 forwards it to the call network gateway. When the call network gateway receives a CM
indication, it forwards it to the answer network gateway which then communicates the CM
indication with the answer modem. The answer modem then responds by transmitting a JM
sequence to the answer network gateway, which is relayed by the answer network gateway to the
5 call modem via the call network gateway. If the call network gateway then receives a CJ
sequence from the call modem, the call negotiator in the call network gateway, initiates operation
in accordance with the V.34 standard, and forwards a CJ sequence to the answer network
gateway. If the JM menu calls for V.34, the call negotiator in the answer network gateway
initiates operation in accordance with the V.34 standard and the call negotiator is terminated. If
10 a standard other than V.34 is called for, the appropriate procedure is invoked, such as those
described previously for V.22 or V.32bis. Next, data rate alignment is achieved by either of two
methods.

15 In a first method for data rate alignment after a V.34 relay connection is established, the
call modem and the answer modem freely negotiate a data rate at each end of the network with
their respective network gateways. Each network gateway forwards a connection rate indication
to the other gateway. Each gateway compares the far end bit rate to the rate transmitted by each
gateway. For example, the call network gateway compares the data rate indication received from
the answer modem gateway to that which it negotiated freely negotiated to with the call modem.
The preferred rate is the minimum of the two rates. Rate renegotiation is invoked if the
20 connection rate at the calling or receiving end differs from the preferred rate, to force the
connection to the desired rate.

25 In an alternate method for V.34 rate synchronization, MP sequences are utilized to
achieve rate synchronization without rate renegotiation. The call modem and the answer modem
independently negotiate with the call network gateway and the answer network gateway
respectively until phase IV of the negotiations is reached. The call network gateway and the
answer network gateway exchange training results in the form of MP sequences when Phase IV
of the independent negotiations is reached to establish the primary and auxiliary data rates. The
call network gateway and the answer network gateway are preferably prevented from relaying
MP sequences to the call modem and the answer modem respectively until the training results
30 for both network gateways and the MP sequences for both modems are available. If symmetric
rate is enforced, the maximum answer data rate and the maximum call data rate of the four MP
sequences are compared. The lower data rate of the two maximum rates is the preferred data rate.
Each network gateway sends the MP sequence with the preferred rate to its respective modem
so that the calling and answer modems operate at the preferred data rate.

35 If asymmetric rates are supported, then the preferred call-answer data rate is the lesser
of the two highest call-answer rates of the four MP sequences. Similarly, the preferred answer-
call data rate is the lesser of the two highest answer-call rates of the four MP sequences. Data
rate capabilities may also need to be modified when the MP sequence are formed so as to be sent

1 to the calling and answer modems. The MP sequence sent to the calling and answer modems, is the logical AND of the data rate capabilities from the four MP sequences.

(V.90 Handshaking Sequence)

5 The V.90 standard utilizes a digital and analog modem pair to transmit modem data over the PSTN line. The V.90 standard utilizes MP sequences to convey training results from a digital to an analog modem, and a similar sequence, using constellation parameters (CP) to convey training results from an analog to a digital modem. Under the V.90 standard, the timeout period is 15 seconds compared to a timeout period of 30 seconds under the V.34 standard. In addition, 10 the analog modems control the handshake timing during training. In an exemplary embodiment, the call modem and the answer modem are the V.90 analog modems. As such the call modem and the answer modem are beyond the control of the network gateways during training. The digital modems only control the timing during transmission of TRN1d, which the digital modem in the network gateway uses to train its echo canceller.

15 When operating in accordance with the V.90 standard, the call negotiator utilizes the V.8 recommendations for initial negotiation. Thus, the initial negotiation of the V.90 relay session is substantially the same as the relay sequence described for V.34 rate synchronization method one and method two with asymmetric rate operation. There are two configurations where V.90 relay may be used. The first configuration is data relay between two V.90 analog modems, i.e. 20 each of the network gateways are configured as V.90 digital modems. The upstream rate between two V.90 analog modems, according to the V.90 standard, is limited to 33,600 bps. Thus, the maximum data rate for an analog to analog relay is 33,600 bps. In accordance with the V.90 standard, the minimum data rate a V.90 digital modem will support is 28,800 bps. Therefore, the connection must be terminated if the maximum data rate for one or both of the 25 upstream directions is less than 28,800 bps, and one or both the downstream direction is in V.90 digital mode. Therefore, the V.34 protocol is preferred over V.90 for data transmission between local and remote analog modems.

30 A second configuration is a connection between a V.90 analog modem and a V.90 digital modem. A typical example of such a configuration is when a user within a packet based PABX system dials out into a remote access server (RAS) or an Internet service provider (ISP) that uses a central site modem for physical access that is V.90 capable. The connection from PABX to the central site modem may be either through PSTN or directly through an ISDN, T1 or E1 interface. Thus the V.90 embodiment should preferably support an analog modem interfacing directly to ISDN, T1 or E1.

35 For an analog to digital modem connection, the connections at both ends of the packet based network should be either digital or analog to achieve proper rate synchronization. The analog modem decides whether to select digital mode as specified in INFO1a, so that INFO1a

1 should be relayed between the calling and answer modem via their respective network gateways before operation mode is synchronized.

5 Upon receipt of an INFO1a signal from the answer modem, the answer network gateway performs a line probe on the signal received from the answer modem to determine whether digital mode can be used. The call network gateway receives an INFO1a signal from the call modem. The call network gateway sends a mode indication to the answer network gateway indicating whether digital or analog will be used and initiates operation in the mode specified in INFO1a. Upon receipt of an analog mode indication signal from the call network gateway, the answer network gateway sends an INFO1a sequence to the answer modem. The answer network gateway then proceeds with analog mode operation. Similarly, if digital mode is indicated and digital mode can be supported by the answer modem, the answer network gateway sends an INFO1a sequence to the answer modem indicating that digital mode is desired and proceeds with digital mode operation.

15 Alternatively, if digital mode is indicated and digital mode can not be supported by the answer modem, the call modem should preferably be forced into analog mode by one of three alternate methods. First, some commercially available V.90 analog modems may revert to analog mode after several retrains. Thus, one method to force the call modem into analog mode is to force retrains until the call modem selects analog mode operation. In an alternate method, the call network gateway modifies its line probe so as to force the call modem to select analog mode. 20 In a third method, the call modem and the answer modem operate in different modes. Under this method if the answer modem can not support a 28,800 bps data rate the connection is terminated.

3. Data Mode Spoofing

25 The jitter buffer 510 may underflow during long delays of data signal packets. Jitter buffer underflow can cause the data pump transmitter 512 to run out of data, and therefore, it is desirable that the jitter buffer 510 be spoofed with bit sequences. Preferably the bit sequences are benign. In the described exemplary embodiment, the specific spoofing methodology is dependent upon the common error mode protocol negotiated by the error control logic of each 30 network gateway.

35 In accordance with V.14 recommendations, the spoofing logic 516 checks for character format and boundary (number of data bits, start bits and stop bits) within the jitter buffer 510. As specified in the V.14 recommendation the spoofing logic 516 must account for stop bits omitted due to asynchronous-to-synchronous conversion. Once the spoofing logic 516 locates the character boundary, ones can be added to spoof the local modem and keep the connection alive. The length of time a modem can be spoofed with ones depends only upon the application program driving the local modem.

1 In accordance with the V.42 recommendations, the spoofing logic 516 checks for HDLC
flag (HDLC frame boundary) within the jitter buffer 510. The basic HDLC structure consists
of a number of frames, each of which is subdivided into a number of fields. The HDLC frame
structure provides for frame labeling and error checking. When a new data transmission is
5 initiated, HDLC preamble in the form of synchronization sequences are transmitted prior to the
binary coded information. The HDLC preamble is modulated bit streams of "01111110 (0x7e)".
The jitter buffer 510 should be sufficiently large to guarantee that at least one complete HDLC
frame is contained within the jitter buffer 510. The default length of an HDLC frame is 132
octets. The V.42 recommendations for error correction of data circuit terminating equipment
10 (DCE) using asynchronous-to-synchronous conversion does not specify a maximum length for
an HDLC frame. However, because the length of the frame affects the overall memory required
to implement the protocol, a information frame length larger than 260 octets is unlikely.

The spoofing logic 516 stores a threshold water mark (with a value set to be
approximately equal to the maximum length of the HDLC frame). Spoofing is preferably
15 activated when the number of packets stored in the jitter buffer 510 drops to the predetermined
threshold level. When spoofing is required, the spoofing logic 516 adds HDLC flags at the frame
boundary as a complete frame is being reassembled and forwarded to the transmit data pump.
This continues until the number of data packets in the jitter buffer 510 exceeds the threshold
level.

20 4. Retrain and Rate Renegotiation

In the described exemplary embodiment, if data rates independently negotiated between
the modems and their respective network gateways are different, the rate negotiator will
renegotiate the data rate by adopting the lowest of the two data rates. The call and answer
25 modems will then undergo retraining or rate renegotiation procedures by their respective network
gateways to establish a new connection at the renegotiated data rate. In addition, rate
synchronization may be lost during a modem communication, requiring modem retraining and
rate renegotiation, due to drift or change in the conditions of the communication channel. When
a retrain occurs, an indication should be forwarded to the network gateway at the end of the
packet based network. The network gateway receiving a retrain indication should initiate retrain
30 with the connected modem to keep data flow in synchronism between the two connections. Rate
synchronization procedures as previously described should be used to maintain data rate
alignment after retrains.

Similarly, rate renegotiation causes both the calling and answer network gateways and
to perform rate renegotiation. However, rate signals or MP (CP) sequences should be exchanged
35 per method two of the data rate alignment as previously discussed for a V.32bis or V.34 rate
synchronization whichever is appropriate.

5. Error Correcting Mode Synchronization

1 Error control (V.42) and data compression (V.42bis) modes should be synchronized at
each end of the packet based network. In a first method, the call modem and the answer modem
independently negotiate an error correction mode with each other on their own, transparent to the
network gateways. This method is preferred for connections wherein the network delay plus
5 jitter is relatively small, as characterized by an overall round trip delay of less than 700 msec.

Data compression mode is negotiated within V.42 so that the appropriate mode
indication can be relayed when the calling and answer modems have entered into V.42 mode.

10 An alternative method is to allow modems at both ends to freely negotiate the error
control mode with their respective network gateways. The network gateways must fully support
all error correction modes when using this method. Also, this method cannot support the
scenario where one modem selects V.14 while the other modem selects a mode other than V.14.
For the case where V.14 is negotiated at both sides of the packet based network, an 8-bit no
parity format is assumed by each respective network gateway and the raw demodulated data bits
are transported there between. With all other cases, each gateway shall extract de-framed (error
15 corrected) data bits and forward them to its counterpart at the opposite end of the network. Flow
control procedures within the error control protocol may be used to handle network delay. The
advantage of this method over the first method is its ability to handle large network delays and
also the scenario where the local connection rates at the network gateways are different.
However, packets transported over the network in accordance with this method must be
guaranteed to be error free. This may be achieved by establishing a connection between the
20 network gateways in accordance with the link access protocol connection for modems (LAPM)

6. Data Pump

25 Preferably, the data exchange includes a modem relay having a data pump for
demodulating modem data signals from a modem for transmission on the packet based network,
and remodulating modem data signal packets from the packet based network for transmission to
a local modem. Similarly, the data exchange also preferably includes a fax relay with a data
pump for demodulating fax data signals from a fax for transmission on the packet based network,
and remodulating fax data signal packets from the packet based network for transmission to a
30 local fax device. The utilization of a data pump in the fax and modem relays to demodulate and
remodulate data signals for transmission across a packet based network provides considerable
bandwidth savings. First, only the underlying unmodulated data signals are transmitted across
the packet based network. Second, data transmission rates of digital signals across the packet
based network, typically 64 kbps is greater than the maximum rate available (typically 33,600
35 bps) for communication over a circuit switched network.

Telephone line data pumps operating in accordance with ITU V series
recommendations for transmission rates of 2400 bps or more typically utilize quadrature
amplitude modulation (QAM). A typical QAM data pump transmitter 600 is shown

1 schematically in FIG. 28. The transmitter input is a serial binary data stream d_n arriving at a rate of R_d bps. A serial to parallel converter 602 groups the input bits into J -bit binary words. A constellation mapper 604 maps each J -bit binary word to a channel symbol from a 2^J element alphabet resulting in a channel symbol rate of $f_s = R_d/J$ baud. The alphabet consists of a pair of real
 5 numbers representing points in a two-dimensional space, called the signal constellation. Customarily the signal constellation can be thought of as a complex plane so that the channel symbol sequence may be represented as a sequence of complex numbers $c_n = a_n + jb_n$. Typically the real part a_n is called the in-phase or I component and the imaginary b_n is called the quadrature or Q component. A nonlinear encoder 605 may be used to expand the constellation points in order to combat the negative effects of companding in accordance with ITU-T G.711 standard.
 10 The I & Q components may be modulated by impulse modulators 606 and 608 respectively and filtered by transmit shaping filters 610 and 612 each with impulse response $g_T(t)$. The outputs of the shaping filters 610 and 612 are called in-phase 610(a) and quadrature 612(a) components of the continuous-time transmitted signal.

15 The shaping filters 610 and 612 are typically lowpass filters approximating the raised cosine or square root of raised cosine response, having a cutoff frequency on the order of at least about $f_s/2$. The outputs 610(a) and 612(a) of the lowpass filters 610 and 612 respectively are lowpass signals with a frequency domain extending down to approximately zero hertz. A local oscillator 614 generates quadrature carriers $\cos(\omega_c t)$ 614(a) and $\sin(\omega_c t)$ 614(b). Multipliers 616 and 618 multiply the filter outputs 610(a) and 612(a) by quadrature carriers $\cos(\omega_c t)$ and $\sin(\omega_c t)$
 20 respectively to amplitude modulate the in-phase and quadrature signals up to the passband of a bandpass channel. The modulated output signals 616(a) and 618(a) are then subtracted in a difference operator 620 to form a transmit output signal 622. The carrier frequency should be greater than the shaping filter cutoff frequency to prevent spectral fold-over.

25 A data pump receiver 630 is shown schematically in FIG. 29. The data pump receiver 630 is generally configured to process a received signal 630(a) distorted by the non-ideal frequency response of the channel and additive noise in a transmit data pump (not shown) in the local modem. An analog to digital converter (A/D) 631 converts the received signal 630(a) from an analog to a digital format. The A/D converter 631 samples the received signal 630(a) at a rate of $f_s = 1/T_s = n_s/T$ which is n_s times the symbol rate $f_s = 1/T$ and is at least twice the highest
 30 frequency component of the received signal 630(a) to satisfy nyquist sampling theory.

35 An echo canceller 634 substantially removes the line echos on the received signal 630(a). Echo cancellation permits a modem to operate in a full duplex transmission mode on a two-line circuit, such as a PSTN. With echo cancellation, a modem can establish two high-speed channels in opposite directions. Through the use of digital-signal-processing circuitry, the modem's receiver can use the shape of the modem's transmitter signal to cancel out the effect of its own transmitted signal by subtracting reference signal and the receive signal 630(a) in a difference operator 633.

1 Multiplier 636 scales the amplitude of echo cancelled signal 633(a). A power estimator
637 estimates the power level of the gain adjusted signal 636(a). Automatic gain control logic
638 compares the estimated power level to a set of predetermined thresholds and inputs a scaling
5 factor into the multiplier 636 that adjusts the amplitude of the echo canceled signal 633(a) to a
level that is within the desired amplitude range. A carrier detector 642 processes the output of
a digital resampler 640 to determine when a data signal is actually present at the input to receiver
630. Many of the receiver functions are preferably not invoked until an input signal is detected.

10 A timing recovery system 644 synchronizes the transmit clock of the remote data pump
transmitter (not shown) and the receiver clock. The timing recovery system 644 extracts timing
information from the received signal, and adjusts the digital resampler 640 to ensure that the
frequency and phase of the transmit clock and receiver clock are synchronized. A phase splitting
fractionally spaced equalizer (PSFSE) 646 filters the received signal at the symbol rate. The
15 PSFSE 646 compensates for the amplitude response and envelope delay of the channel so as to
minimize inter-symbol interference in the received signal. The frequency response of a typical
channel is inexact so that an adaptive filter is preferable. The PSFSE 646 is preferably an
adaptive FIR filter that operates on data signal samples spaced by T/n_0 and generates digital
signal output samples spaced by the period T . In the described exemplary embodiment $n_0=3$.

20 The PSFSE 646 outputs a complex signal which multiplier 650 multiplies by a locally
generated carrier reference 652 to demodulate the PSFSE output to the baseband signal 650(a).
The received signal 630(a) is typically encoded with a non-linear operation so as to reduce the
quantization noise introduced by companding in accordance with ITU-T G.711. The baseband
signal 650(a) is therefore processed by a non-linear decoder 654 which reverses the non-linear
encoding or warping. The gain of the baseband signal will typically vary upon transition from
a training phase to a data phase because modem manufacturers utilize different methods to
25 compute a scale factor. The problem that arises is that digital modulation techniques such as
quadrature amplitude modulation (QAM) and pulse amplitude modulation (PAM) rely on precise
gain (or scaling) in order to achieve satisfactory performance. Therefore, a scaling error
compensator 656 adjusts the gain of the receiver to compensate for variations in scaling. Further,
a slicer 658 then quantizes the scaled baseband symbols to the nearest ideal constellation points,
which are the estimates of the symbols from the remote data pump transmitter (not shown). A
30 decoder 659 converts the output of slicer 658 into a digital binary stream.

35 During data pump training, known transmitted training sequences are transmitted by a
data pump transmitter in accordance with the applicable ITU-T standard. An ideal reference
generator 660, generates a local replica of the constellation point 660(a). During the training
phase a switch 661 is toggled to connect the output 660(a) of the ideal reference generator 660
to a difference operator 662 that generates a baseband error signal 662(a) by subtracting the ideal
constellation sequence 660(a) and the baseband equalizer output signal 650(a). A carrier phase
generator 664 uses the baseband error signal 662(a) and the baseband equalizer output signal
650(a) to synchronize local carrier reference 666 with the carrier of the received signal 630(a)

1 During the data phase the switch 661 connects the output 658(a) of the slicer to the input of
difference operator 662 that generates a baseband error signal 662(a) in the data phase by
subtracting the estimated symbol output by the slicer 658 and the baseband equalizer output
5 signal 650(a). It will be appreciated by one of skill that the described receiver is one of several
approaches. Alternate approaches in accordance with ITU-T recommendations may be readily
substituted for the described data pump. Accordingly, the described exemplary embodiment of
the data pump is by way of example only and not by way of limitation.

a. Timing Recovery System

10 Timing recovery refers to the process in a synchronous communication system whereby
timing information is extracted from the data being received. In the context of a modem
connection in accordance with an exemplary embodiment of the present invention, each modem
is coupled to a signal processing system, which for the purposes of explanation is operating in
a network gateway, either directly or through a PSTN line. In operation, each modem establishes
15 a modem connection with its respective network gateway, at which point, the modems begin
relaying data signals across a packet based network. The problem that arises is that the clock
frequencies of the modems are not identical to the clock frequencies of the data pumps operating
in their respective network gateways. By design, the data pump receiver in the network gateway
should sample a received signal of symbols in synchronism with the transmitter clock of the
modem connected locally to that gateway in order to properly demodulate the transmitted signal.

20 A timing recovery system can be used for this purpose. Although the timing recovery
system is described in the context of a data pump within a signal processing system with the
packet data modem exchange invoked, those skilled in the art will appreciate that the timing
recovery system is likewise suitable for various other applications in various other telephony and
telecommunications applications, including fax data pumps. Accordingly, the described
25 exemplary embodiment of the timing recovery system in a signal processing system is by way
of example only and not by way of limitation.

30 A block diagram of a timing recovery system is shown in FIG. 30. In the described
exemplary embodiment, the digital resampler 640 resamples the gain adjusted signal 636(a)
output by the AGC (see FIG. 29). A timing error estimator 670 provides an indication of
whether the local timing or clock of the data pump receiver is leading or lagging the timing or
clock of the data pump transmitter in the local modem. As is known in the art, the timing error
estimator 670 may be implemented by a variety of techniques including that proposed by Godard.
The A/D converter 631 of the data pump receiver (see FIG. 29) samples the received signal
35 630(a) at a rate of f_0 which is an integer multiple of the symbol rate $f_s = 1/T$ and is at least twice
the highest frequency component of the received signal 630(a) to satisfy nyquist sampling theory.
The samples are applied to an upper bandpass filter 672 and a lower bandpass filter 674. The
upper bandpass filter 672 is tuned to the upper bandedge frequency $f_u = f_c + 0.5f_s$ and the lower
bandpass filter 674 is tuned to the lower bandedge frequency $f_l = f_c - 0.5f_s$ where f_c is the carrier

1 frequency of the QAM signal. The bandwidth of the filters 672 and 674 should be reasonably
narrow, preferably on the order of 100 Hz for a $f_s = 2400$ baud modem. Conjugate logic 676
takes the complex conjugate of complex output of the lower bandpass filter. Multiplier 678
multiplies the complex output of the upper bandpass filter 672(a) by the complex conjugate of
5 the lower bandpass filter to form a cross-correlation between the output of the two filters (672
and 674). The real part of the correlated symbol is discarded by processing logic 680, and a
sampler 681 samples the imaginary part of the resulting cross-correlation at the symbol rate to
provide an indication of whether the timing phase error is leading or lagging.

10 In operation, a transmitted signal from a remote data pump transmitter (not shown) $g(t)$
is made to correspond to each data character. The signal element has a bandwidth approximately
equal to the signaling rate f_s . The modulation used to transmit this signal element consists of
multiplying the signal by a sinusoidal carrier of frequency f_c which causes the spectrum to be
translated to a band around frequency f_c . Thus, the corresponding spectrum is bounded by
frequencies $f_1 = f_c - 0.5f_s$ and $f_2 = f_c + 0.5f_s$, which are known as the bandedge frequencies.
15 Reference for more detailed information may be made to "Principles of Data Communication"
by R. W. Lucky, J. Salz and E. J. Weldon, Jr., McGraw-Hill Book Company, pages 50-51.

In practice it has been found that additional filtering is required to reduce symbol clock
jitter, particularly when the signal constellation contains many points. Conventionally a loop
filter 682 filters the timing recovery signal to reduce the symbol clock jitter. Traditionally the
20 loop filter 682 is a second order infinite impulse response (IIR) type filter, whereby the second
order portion tracks the offset in clock frequency and the first order portion tracks the offset in
phase. The output of the loop filter drives clock phase adjuster 684. The clock phase adjuster
controls the digital sampling rate of digital resampler 640 so as to sample the received symbols
in synchronism with the transmitter clock of the modem connected locally to that gateway.
25 Typically, the clock phase adjuster 684 utilizes a poly-phase interpolation algorithm to digitally
adjust the timing phase. The timing recovery system may be implemented in either analog or
digital form. Although digital implementations are more prevalent in current modem design an
analog embodiment may be realized by replacing the clock phase adjuster with a VCO.

30 The loop filter 682 is typically implemented as shown in FIG. 31. The first order portion
of the filter controls the adjustments made to the phase of the clock (not shown). A multiplier
688 applies a first order adjustment constant α to advance or retard the clock phase adjustment.
Typically the constant α is empirically derived via computer simulation or a series of simple
experiments with a telephone network simulator. Generally α is dependent upon the gain and
the bandwidth of the upper and lower filters in the timing error estimator, and is generally
35 optimized to reduce symbol clock jitter and control the speed at which the phase is adjusted. The
structure of the loop filter 682 may include a second order component 690 that estimates the
offset in clock frequency. The second order portion utilizes an accumulator 692 in a feedback
loop to accumulate the timing error estimates. A multiplier 694 is used to scale the accumulated
timing error estimate by a constant β . Typically, the constant β is empirically derived based on

1 the amount of feedback that will cause the system to remain stable. Summer 695 sums the scaled
accumulated frequency adjustment 694(a) with the scaled phase adjustment 688(a). A
disadvantage of conventional designs which include a second order component 690 in the loop
filter 682 is that such second order components 690 are prone to instability with large
5 constellation modulations under certain channel conditions.

An alternative digital implementation eliminates the loop filter. Referring to FIG. 32 a
hard limiter 695 and a random walk filter 696 are coupled to the output of the timing error
estimator 680 to reduce timing jitter. The hard limiter 695 provides a simple automatic gain
control action that keeps the loop gain constant independent of the amplitude level of the input
10 signal. The hard limiter 695 assures that timing adjustments are proportional to the timing of the
data pump transmitter of the local modem and not the amplitude of the received signal. The
random walk filter 696 reduces the timing jitter induced into the system as disclosed in
"Communication System Design Using DSP Algorithms", S. Tretter, p. 132, Plenum Press, NY.,
1995, the contents of which is hereby incorporated by reference as through set forth in full herein.
15 The random walk filter 696 acts as an accumulator, summing a random number of adjustments
over time. The random walk filter 696 is reset when the accumulated value exceeds a positive
or negative threshold. Typically, the sampling phase is not adjusted so long as the accumulator
output remains between the thresholds, thereby substantially reducing or eliminating incremental
positive adjustments followed by negative adjustments that otherwise tend to not accumulate.

20 Referring to FIG. 33 in an exemplary embodiment of the present invention, the multiplier
688 applies the first order adjustment constant α to the output of the random walk filter to
advance or retard the estimated clock phase adjustment. In addition, a timing frequency offset
compensator 697 is coupled to the timing recovery system via switches 698 and 699 to preferably
provide a fixed dc component to compensate for clock frequency offset present in the received
25 signal. The exemplary timing frequency offset compensator preferably operates in phases. A
frequency offset estimator 700 computes the total frequency offset to apply during an estimation
phase and incremental logic 701, incrementally applies the offset estimate in linear steps during
the application phase. Switch control logic 702 controls the toggling of switches 698 and 699
during the estimation and application phases of compensation adjustment. Unlike the second
order component 690 of the conventional timing recovery loop filter disclosed in FIG. 31, the
30 described exemplary timing frequency offset compensator 697 is an open loop design such that
the second order compensation is fixed during steady state. Therefore, switches 698 and 699
work in opposite cooperation when the timing compensation is being estimated and when it is
being applied.

35 During the estimation phase, switch control logic 702 closes switch 698 thereby coupling
the timing frequency offset compensator 697 to the output of the random walk filter 696, and
opens switch 699 so that timing adjustments are not applied during the estimation phase. The
frequency offset estimator 700 computes the timing frequency offset during the estimation phase
over K symbols in accordance with the block diagram shown in FIG. 34. An accumulator 703

1 accumulates the frequency offset estimates over K symbols. A multiplier 704 is used to average
the accumulated offset estimate by applying a constant γ/K . Typically the constant γ is
empirically derived and is preferably in the range of about 0.5-2. Preferably K is as large as
possible to improve the accuracy of the average. K is typically greater than about 500 symbols
5 and less than the recommended training sequence length for the modem in question. In the
exemplary embodiment the first order adjustment constant α is preferably in the range of about
100-300 part per million (ppm). The timing frequency offset is preferably estimated during the
timing training phase (timing tone) and equalizer training phase based on the accumulated
adjustments made to the clock phase adjuster 684 over a period of time.

10 During steady state operation when the timing adjustments are applied, switch control
logic 702 opens switch 698 decoupling the timing frequency offset compensator 697 from the
output of the random walk filter, and closes switch 699 so that timing adjustments are applied
by summer 705. After K symbols of a symbol period have elapsed and the frequency offset
15 compensation is computed, the incremental logic 701 preferably applies the timing frequency
offset estimate in incremental linear steps over a period of time to avoid large sudden adjustments
which may throw the feedback loop out of lock. This is the transient phase. The length of time
over which the frequency offset compensation is incrementally applied is empirically derived,
and is preferably in the range of about 200-800 symbols. After the incremental logic 701 has
20 incrementally applied the total timing frequency offset estimate computed during the estimate
phase, a steady state phase begins where the compensation is fixed. Relative to conventional
second order loop filters, the described exemplary embodiment provides improved stability and
robustness.

b. Multipass Training

25 Data pump training refers to the process by which training sequences are utilized to train
various adaptive elements within a data pump receiver. During data pump training, known
transmitted training sequences are transmitted by a data pump transmitter in accordance with the
applicable ITU-T standard. In the context of a modem connection in accordance with an
exemplary embodiment of the present invention, the modems (see FIG. 24) are coupled to a
signal processing system, which for the purposes of explanation is operating in a network
30 gateway, either directly or through a PSTN line. In operation, the receive data pump operating
in each network gateway of the described exemplary embodiment utilizes PSFSE architecture.
The PSFSE architecture has numerous advantages over other architectures when receiving QAM
signals. However, the PSFSE architecture has a slow convergence rate when employing the least
mean square (LMS) stochastic gradient algorithm. This slow convergence rate typically prevents
35 the use of PSFSE architecture in modems that employ relatively short training sequences in
accordance with common standards such as V.29. Because of the slow convergence rate, the
described exemplary embodiment re-processes blocks of training samples multiple times (multi-
pass training).

1 Although the method of performing multi-pass training is described in the context of a
signal processing system with the packet data exchange invoked, those skilled in the art will
appreciate that multi-pass training is likewise suitable for various other telephony and
telecommunications applications. Accordingly, the described exemplary method for multi-pass
5 training in a signal processing system is by way of example only and not by way of limitation.

In an exemplary embodiment the data pump receiver operating in the network gateway
stores the received QAM samples of the modem's training sequence in a buffer until N symbols
have been received. The PSFSE is then adapted sequentially over these N symbols using a LMS
10 algorithm to provide a coarse convergence of the PSFSE. The coarsely converged PSFSE (i.e.
with updated values for the equalizer taps) returns to the start of the same block of training
samples and adapts a second time. This process is repeated M times over each block of training
samples. Each of the M iterations provides a more precise or finer convergence until the PSFSE
is completely converged.

15 c. Scaling Error Compensator

Scaling error compensation refers to the process by which the gain of a data pump
receiver (fax or modem) is adjusted to compensate for variations in transmission channel
conditions. In the context of a modem connection in accordance with an exemplary embodiment
of the present invention, each modem is coupled to a signal processing system, which for the
20 purposes of explanation is operating in a network gateway, either directly or through a PSTN
line. In operation, each modem communicates with its respective network gateway using digital
modulation techniques. The problem that arises is that digital modulation techniques such as
QAM and pulse amplitude modulation (PAM) rely on precise gain (or scaling) in order to
achieve satisfactory performance. In addition, transmission in accordance with the V.34
25 recommendations typically includes a training phase and a data phase whereby a much smaller
constellation size is used during the training phase relative to that used in the data phase. The
V.34 recommendation, requires scaling to be applied when switching from the smaller
constellation during the training phase into the larger constellation during the data phase.

The scaling factor can be precisely computed by theoretical analysis, however, different
30 manufacturers of V.34 systems (modems) tend to use slightly different scaling factors. Scaling
factor variation (or error) from the predicted value may degrade performance until the PSFSE
compensates for the variation in scaling factor. Variation in gain due to transmission channel
conditions is compensated by an initial gain estimation algorithm (typically consisting of a
simple signal power measurement during a particular signaling phase) and an adaptive equalizer
35 during the training phase. However, since a PSFSE is preferably configured to adapt very slowly
during the data phase, there may be a significant number of data bits received in error before the
PSFSE has sufficient time to adapt to the scaling error.

1 It is, therefore, desirable to quickly reduce the scaling error and hence minimize the
number of potential erred bits. A scaling factor compensator can be used for this purpose.
Although the scaling factor compensator is described in the context of a signal processing system
with the packet data modem exchange invoked, those skilled in the art will appreciate that the
5 preferred scaling factor compensator is likewise suitable for various other telephony and
telecommunications applications. Accordingly, the described exemplary embodiment of the
scaling factor compensator in a signal processing system is by way of example only and not by
way of limitation.

10 FIG. 35 shows a block diagram of an exemplary embodiment of the scaling error
compensator in a data pump receiver 630 (see FIG. 29). In an exemplary embodiment, scaling
error compensator 708 computes the gain adjustment of the data pump receiver. Multiplier 710
adjusts a nominal scaling factor 712 (the scaling error computed by the data pump manufacturer)
by the gain adjustment as computed by the scaling error compensator 708. The combined scale
factor 710(a) is applied to the incoming symbols by multiplier 714. A slicer 716 quantizes the
15 scaled baseband symbols to the nearest ideal constellation points, which are the estimates of the
symbols from the remote data pump transmitter.

The scaling error compensator 708 preferably includes a divider 718 which estimates the
gain adjustment of the data pump receiver by dividing the expected magnitude of the received
symbol 716(a) by the actual magnitude of the received symbol 716(b). In the described
20 exemplary embodiment the magnitude is defined as the sum of squares between real and
imaginary parts of the complex symbol. The expected magnitude of the received symbol is the
output 716(a) of the slicer 716 (i.e. the symbol quantized to the nearest ideal constellation point)
whereas the magnitude of the actual received symbol is the input 716(b) to the slicer 716. In the
case where a Viterbi decoder performs the error-correction of the received, noise-disturbed signal
25 (as for V.34), the output of the slicer may be replaced by the first level decision of the Viterbi
decoder.

The statistical nature of noise is such that large spikes in the amplitude of the received
signal will occasionally occur. A large spike in the amplitude of the received signal may result
in an erroneously large estimate of the gain adjustment of the data pump receiver. Typically,
30 scaling is applied in a one to one ratio with the estimate of the gain adjustment, so that large
scaling factors may be erroneously applied when large amplitude noise spikes are received. To
minimize the impact of large amplitude spikes and improve the accuracy of the system, the
described exemplary scaling error compensator 708 further includes a non-linear filter in the
form of a hard-limiter 720 which is applied to each estimate 718(a). The hard limiter 720 limits
35 the maximum adjustment of the scaling value. The hard limiter 720 provides a simple automatic
control action that keeps the loop gain constant independent of the amplitude of the input signal
so as to minimize the negative effects of large amplitude noise spikes. In addition, averaging
logic 722 computes the average gain adjustment estimate over a number (N) of symbols in the

1 data phase prior to adjusting the nominal scale factor 710. As will be appreciated by those of skill in the art, other non-linear filtering algorithms may also be used in place of the hard-limiter.

5 Alternatively, the accuracy of the scaling error compensation may be further improved by estimating the averaged scaling adjustment twice and applying that estimate in two steps. A large hard limit value (typically 1 ± 0.25) is used to compute the first average scaling adjustment. The initial prediction provides an estimate of the average value of the amplitude of the received symbols. The unpredictable nature of the amplitude of the received signal requires the use of a large initial hard limit value to ensure that the true scaling error is included in the initial estimate of the average scaling adjustment. The estimate of the average value of the amplitude of the received symbols is used to calibrate the limits of the scaling adjustment. The average scaling adjustment is then estimated a second time using a lower hard limit value and then applied to the nominal scale factor 712 by multiplier 710.

10 In most modem specifications, such as the V.34 standards, there is a defined signaling period (B1 for V.34) after transition into data phase where the data phase constellation is transmitted with signaling information to flush the receiver pipeline (i.e. Viterbi decoder etc.) prior to the transmission of actual data. In an exemplary embodiment this signaling period may be used to make the scaling adjustment such that any scaling error is compensated for prior to actual transfer of data.

20 d. Non-Linear Decoder

25 In the context of a modem connection in accordance with an exemplary embodiment of the present invention, each modem is coupled to a signal processing system, which for the purposes of explanation is operating in a network gateway, either directly or through a PSTN line. In operation, each modem communicates with its respective network gateway using digital modulation techniques. The international telecommunications union (ITU) has promulgated standards for the encoding and decoding of digital data in ITU-T Recommendation G.711 (ref. G.711) which is incorporated herein by reference as if set forth in full. The encoding standard specifies that a nonlinear operation (companding) be performed on the analog data signal prior to quantization into seven bits plus a sign bit. The companding operation is a monatomic invertable function which reduces the higher signal levels. At the decoder, the inverse operation (expanding) is done prior to analog reconstruction. The companding / expanding operation quantizes the higher signal values more coarsely. The companding / expanding operation, is suitable for the transmission of voice signals but introduces quantization noise on data modem signals. The quantization error (noise) is greater for the outer signal levels than the inner signal levels.

35 The ITU-T Recommendation V.34 describes a mechanism whereby (ref. V.34) the uniform signal is first expanded (ref. BETTS) to space the outer points farther apart than the inner points before G.711 encoding and transmission over the PCM link. At the receiver, the

1 inverse operation is applied after G.711 decoding. The V.34 recommended expansion / inverse
operation yields a more uniform signal to noise ratio over the signal amplitude. However, the
inverse operation specified in the ITU-T Recommendation V.34 requires a complex receiver
calculation. The calculation is computationally intensive, typically requiring numerous machine
5 cycles to implement.

It is, therefore, desirable to reduce the number of machine cycles required to compute the
inverse to within an acceptable error level. A simplified nonlinear decoder can be used for this
purpose. Although the nonlinear decoder is described in the context of a signal processing
system with the packet data modem exchange invoked, those skilled in the art will appreciate that
10 the nonlinear decoder is likewise suitable for various other telephony and telecommunications
application. Accordingly, the described exemplary embodiment of the nonlinear decoder in a
signal processing system is by way of example only and not by way of limitation.

Conventionally, iteration algorithms have been used to compute the inverse of the G.711
15 nonlinear warping function. Typically, iteration algorithms generate an initial estimate of the
input to the nonlinear function and then compute the output. The iteration algorithm compares
the output to a reference value and adjusts the input to the nonlinear function. A commonly used
adjustment is the successive approximation wherein the difference between the output and the
reference function is added to the input. However, when using the successive approximation
20 technique, up to ten iterations may be required to adjust the estimated input of the nonlinear
warping function to an acceptable error level, so that the nonlinear warping function must be
evaluated ten times. The successive approximation technique is computationally intensive,
requiring significant machine cycles to converge to an acceptable approximation of the inverse
of the nonlinear warping function. Alternatively, a more complex warping function is a linear
Newton Rhapsion iteration. Typically the Newton Rhapsion algorithm requires three evaluations
25 to converge to an acceptable error level. However, the inner computations for the Newton
Rhapsion algorithm are more complex than those required for the successive approximation
technique. The Newton Rhapsion algorithm utilizes a computationally intensive iteration loop
wherein the derivative of the nonlinear warping function is computed for each approximation
iteration, so that significant machine cycles are required to conventionally execute the Newton
Rhapsion algorithm.

30 An exemplary embodiment of the present invention modifies the successive
approximation iteration. A presently preferred algorithm computes an approximation to the
derivative of the nonlinear warping function once before the iteration loop is executed and uses
the approximation as a scale factor during the successive approximation iterations. The described
exemplary embodiment converges to the same acceptable error level as the more complex
35 conventional Newton-Rhapsion algorithm in four iterations. The described exemplary
embodiment further improves the computational efficiency by utilizing a simplified
approximation of the derivative of the nonlinear warping function.

In operation, development of the described exemplary embodiment proceeds as follows with a warping function defined as:

$$w(v) = \frac{\Theta(v)}{6} + \frac{\Theta(v)^2}{120}$$

the V.34 nonlinear decoder can be written as

$$Y = X(1 + w(\|X\|^2))$$

taking the square of the magnitude of both sides yields,

$$Y^2 = \|X\|^2 (1 + w(\|X\|^2))^2$$

The encoder notation can then be simplified with the following substitutions

$$Y_r = \|Y\|^2, X_r = \|X\|^2$$

and write the V.34 nonlinear encoder equation in the canonical form $G(x)=0$.

$$X_r(1 + w(X_r))^2 - Y_r = 0$$

The Newton-Rhapson iteration is a numerical method to determine X that results in an iteration of the form:

$$X_{n+1} = X_n - \frac{G(X_n)}{G'(X_n)}$$

where G' is the derivative and the substitution iteration results when G' is set equal to one.

The computational complexity of the Newton-Rhapson algorithm is thus paced by the derivation of the derivative G' , which conventionally is related to X , so that the mathematical instructions saved by performing fewer iterations are offset by the instructions required to calculate the derivative and perform the divide. Therefore, it would be desirable to approximate the derivative G' with a term that is the function of the input Y , so that $G(x)$ is a monotonic function and $G'(x)$ can be expressed in terms of $G(x)$. Advantageously, if the steps in the iteration are small, then $G'(x)$ will not vary greatly and can be held constant over the iteration. A series of simple experiments yields the following approximation of $G'(x)$ where α is an experimentally derived scaling factor.

$$G' = \frac{1+Y_r}{\alpha}$$

1 The approximation for G' converges to an acceptable error level in a minimum number of steps, typically one more iteration than the full linear Newton-Rhapson algorithm. A single divide before the iteration loop computes the quantity

$$5 \quad \frac{1}{G'} = \frac{\alpha}{1+Yr}$$

 The error term is multiplied by $1/G'$ in the successive iteration loop. It will be appreciated by one of skill in the art that further improvements in the speed of convergence are possible with the "Generalized Newton-Rhapson" class of algorithms. However, the inner loop computations for this class of algorithm are quite complex.

10 Advantageously, the described exemplary embodiment does not expand the polynomial because the numeric quantization on a store in a sixteen bit machine may be quite significant for the higher order polynomial terms. The described exemplary embodiment organizes the inner loop computations to minimize the effects of truncation and the number of instructions required for execution. Typically the inner loop requires eighteen instructions and four iterations to converge to within two bits of the actual value which is within the computational roundoff noise of a sixteen bit machine.

D. Human Voice Detector

20 In a preferred embodiment of the present invention, a signal processing system is employed to interface telephony devices with packet based networks. Telephony devices include, by way of example, analog and digital phones, ethernet phones, Internet Protocol phones, fax machines, data modems, cable voice modems, interactive voice response systems, PBXs, key systems, and any other conventional telephony devices known in the art. In the described exemplary embodiment the packet voice exchange is common to both the voice mode and the voiceband data mode. In the voiceband data mode, the network VHD invokes the packet voice exchange for transparently exchanging data without modification (other than packetization) between the telephony device or circuit switched network and the packet based network. This is typically used for the exchange of fax and modem data when bandwidth concerns are minimal as an alternative to demodulation and remodulation.

30 During the voiceband data mode, the human voice detector service is also invoked by the resource manager. The human voice detector monitors the signal from the near end telephony device for voice. The described exemplary human voice detector estimates pitch period of an incoming telephony signal and compares the pitch period of said telephony signal to a plurality of thresholds to identify active voice samples. This approach is substantially independent of the amplitude of the spoken utterance, so that whispered or shouted utterance may be accurately identified as active voice samples. In the event that voice is detected by the human voice detector, an event is forwarded to the resource manager which, in turn, causes the resource manager to

1 terminate the human voice detector service and invoke the appropriate services for the voice mode (i.e., the call discriminator, the packet tone exchange, and the packet voice exchange).

5 Although a preferred embodiment is described in the context of a signal processing system for telephone communications across the packet based network, it will be appreciated by those skilled in the art that the voice detector is likewise suitable for various other telephony and telecommunications application. Accordingly, the described exemplary embodiment of the voice detector in a signal processing system is by way of example only and not by way of limitation.

10 There are a variety of encoding methods known for encoding voice. Most frequently, voice is modeled on a short-time basis as the response of a linear system excited by a periodic impulse train for voiced sounds or random noise for the unvoiced sounds. Conventional human voice detectors typically monitor the power level of the incoming signal to make a voice / machine decision. Typically, if the power level of the incoming signal is above a predetermined
15 threshold, the sequence is typically declared voice. The performance of such conventional voice detectors may be degraded by the environment, in that a very soft spoken whispered utterance will have a very different power level from a loud shout. If the threshold is set at too low a level, noise will be declared voice, whereas if the threshold is set at too high a level a soft spoken voice segment will be incorrectly marked as inactive.

20 Alternatively, voice may generally be classified as voiced if a fundamental frequency is imported to the air stream by the vocal cords of the speaker. In such case, the frequency of a voice segment is typically highly periodic at around the pitch frequency. The determination as to whether a voice segment is voiced or unvoiced, and the estimation of the fundamental frequency can be obtained in a variety of ways known in the art such as pitch detection
25 algorithms. In the described exemplary embodiment, the human voice detector calculates an autocorrelation function for the incoming signal. An autocorrelation function for a voice segment demonstrates local peaks with a periodicity in proportion to the pitch period. The human voice detector service utilizes this feature in conjunction with power measurements to distinguish voice signals from modem signals. It will be appreciated that other pitch detection algorithms known in the art can be used as well.

30 Referring to FIG. 36, in the described exemplary embodiment, a power estimator 730 estimates the power level of the incoming signal. Autocorrelation logic 732 computes an autocorrelation function for an input signal to assist in the voice/machine decision. Autocorrelation, as is known in the art, involves correlating a signal with itself. A correlation
35 function shows how similar two signals are, and how long the signals remain similar when one is shifted with respect to the other. Periodic signals go in and out of phase as one is shifted with respect to the other, so that a periodic signal will show strong correlation at shifts where the peaks coincide. Thus, the autocorrelation of a periodic signal is itself a periodic signal, with a period equal to the period of the original signal.

1 The autocorrelation calculation computes the autocorrelation function over an interval of 360 samples with the following approach:

$$R[k] = \sum_{n=0}^{N-k-1} x[n]x[n+k]$$

5 where $N=360$, $k=0,1,2,\dots,179$.

10 A pitch tracker 734 estimates the period of the computed autocorrelation function. Framed based decision logic 736 analyzes the estimated power level 730a, the autocorrelation function 732a and the periodicity 734a of the incoming signal to execute a frame based voice/machine decision according to a variety of factors. For example, the energy of the input signal should be above a predetermined threshold level, preferably in the range of about -45 to -55 dBm, before the frame based decision logic 736 declares the signal to be voice. In addition, the typical pitch period of a voice segment should be in the range of about 60-400 Hz, so that the autocorrelation function should preferably be periodic with a period in the range of about 60-400 Hz before the frame based decision logic 736 declares a signal as active or containing voice.

15 The amplitude of the autocorrelation function is a maximum for $R[0]$, i.e. when the signal is not shifted relative to itself. Also, for a periodic voice signal, the amplitude of the autocorrelation function with a one period shift (i.e. $R[\text{pitch period}]$) should preferably be in the range of about 0.25-0.40 of the amplitude of the autocorrelation function with no shift (i.e. $R[0]$). Similarly, modem signaling may involve certain DTMF or MF tones, in this case the signals are highly correlated, so that if the largest peak in the amplitude of the autocorrelation function after $R[0]$ is relatively close in magnitude to $R[0]$, preferably in the range of about 0.75-0.90 $R[0]$, the frame based decision logic 736 declares the sequence as inactive or not containing voice.

25 Once a decision is made on the current frame as to voice or machine, final decision logic 738 compares the current frame decision with the two adjacent frame decisions. This check is known as backtracking. If a decision conflicts with both adjacent decisions it is flipped, i.e. voice decision turned to machine and vice versa.

30 Although a preferred embodiment of the present invention has been described, it should not be construed to limit the scope of the appended claims. For example, the present invention can be implemented by both a software embodiment or a hardware embodiment. Those skilled in the art will understand that various modifications may be made to the described embodiment. Moreover, to those skilled in the various arts, the invention itself herein will suggest solutions to other tasks and adaptations for other applications. It is therefore desired that the present embodiments be considered in all respects as illustrative and not restrictive, reference being made to the appended claims rather than the foregoing description to indicate the scope of the invention.

1 WHAT IS CLAIMED IS:

1. A signal processing system, comprising:
a voice exchange capable of exchanging voice signals between a network line and
5 a packet based network; and
a full duplex data exchange capable of exchanging data signals from the network
line with demodulated data signals from the packet based network.

2. The signal processing system of claim 1 further comprising a call discriminator
capable of discriminating between the voice signals and the data signals from the network line,
10 the voice exchange being enabled for the voice signals and the data exchange being enabled for
the data signals.

3. The signal processing system of claim 1 wherein the data signals from the network
line are modulated by a voiceband carrier, and the data exchange comprises a data pump capable
15 of demodulating the data signals from the network line for transmission on the packet based
network and remodulating the data signals from the packet based network with the voiceband
carrier for transmission on the network line.

4. The signal processing system of claim 3 wherein the data exchange comprises a
jitter buffer capable of receiving packets of the data signals of varying delay from the packet
20 based network and compensating for the delay variation of the data signal packets.

5. The signal processing system of claim 4 wherein the jitter buffer outputs an
isochronous stream of the received data signals.

6. The signal processing system of claim 4 wherein the data pump transmits the
received data signals to the network line at a transmit rate.

7. The signal processing system of claim 6 wherein the jitter buffer compensates for
the delay variation of the data signal packets by holding a number of the received data signals,
and wherein the data exchange further comprises a clock synchronizer which adaptively adjusts
30 the transmit rate of the data pump in response the number of the received data signals in the jitter
buffer.

8. The signal processing system of claim 6 wherein the jitter buffer compensates for
the delay variation of the data signal packets by holding a number of the received data signals,
and wherein the data exchange further comprises spoof logic which provides spoof data to the
data pump when the number of the received data signals held in the jitter buffer is below a
35 threshold.

1 9. The signal processing system of claim 1 wherein the voice exchange comprises a jitter buffer capable of receiving packets of the voice signals of varying delay from the packet based network and compensating for the delay variation of the voice signal packets.

5 10. The signal processing system of claim 9 wherein the jitter buffer outputs an isochronous stream of the received voice signals.

 11. The signal processing system of claim 9 wherein the jitter buffer comprises a voice queue which buffers the received voice signals for a holding time, and a voice synchronizer which adaptively adjusts the holding time of the voice queue.

10 12. The signal processing system of claim 11 further comprising a tone exchange capable of exchanging DTMF signals between the network line and the packet based network, the DTMF exchange comprising a DTMF queue capable of buffering packets of the DTMF signals from the packet based network, and a tone generator which generates a DTMF tone responsive to the buffered DMTF signals, the DMTF queue outputting a signal to the voice synchronizer to suppress the buffered voice signals when the DTMF signals are in the DTMF queue.

15 13. The signal processing system of claim 1 wherein the voice exchange comprises a voice decoder capable of decoding packets of the voice signals from the packet based network for transmission to the network line, a voice activity detector which detects the voice signals without speech, and a comfort noise generator which inserts comfort noise in place of the voice signals without speech.

20 14. The signal processing system of claim 13 wherein the voice exchange further comprises a comfort noise estimator which generates comfort noise parameters from at least a portion of the voice signals without speech, the comfort noise generator being responsive to the comfort noise parameters.

25 15. The signal processing system of claim 1 wherein the voice exchange comprises a voice decoder capable of decoding packets of the voice signals from the packet based network for transmission to the network line, a voice activity detector which detects lost voice signals, and a lost packet recovery engine which processes the voice signals to compensate for the lost voice signals.

30 16. The signal processing system of claim 1 wherein the voice exchange comprises a voice encoder capable of encoding the voice signals from the network line for transmission on the packet based network, and a voice activity detector which suppresses the voice signals without speech.

1 17. The signal processing system of claim 16 further comprising a comfort noise estimator which generates comfort noise parameters when the voice activity detector suppresses the voice signals without speech.

5 18. The signal processing system of claim 1 wherein the voice exchange further comprises a decoder capable of decoding packets of the voice signals from the packet based network, and an echo canceller capable of cancelling decoded voice signal echos on incoming voice signals from the network line.

10 19. The signal processing system of claim 18 further comprising a non-linear processor which mutes the incoming voice signals when the incoming voice signals do not comprise speech and the echo canceller detects the decoded voice signals with speech.

15 20. The signal processing system of claim 1 wherein the voice exchange comprises a voice encoder capable of encoding the voice signals from the network line into voice signal packets for the packet based network.

20 21. The signal processing system of claim 20 further comprising a tone exchange comprising a DTMF detector capable of detecting a DTMF signal from the network line and generating a DTMF packet for the packet based network in response to the DTMF signal, the DTMF detector muting the voice signal packets when a DTMF signal is detected.

25 22. The signal processing system of claim 1 further comprising a fax exchange capable of exchanging fax signals from the network line with demodulated fax signals from the packet based network

30 23. The signal processing system of claim 22 wherein the fax signals from the network line are modulated by a voiceband carrier, and the fax exchange comprises a data pump capable of demodulating the fax signals from the network line for transmission on the packet based network, and remodulating the fax signals from the packet based network with the voiceband carrier for transmission on the network line.

35 24. The signal processing system of claim 22 wherein the call discriminator is capable of discriminating the fax signals from the network line, the fax exchange being enabled for the fax signals.

 25. The signal processing system of claim 1 wherein the data exchange comprises a rate synchronizer capable of receiving data rate codes from the packet based network and setting a data rate of a telephony device coupled to the network line in response to the received data rate codes.

1 26. A signal processing system, comprising:
 a voice exchange capable of exchanging voice signals between a first telephony
device and a packet based network;
 a full duplex data exchange capable of exchanging data signals from a second
5 telephony device with demodulated data signals from the packet based network; and
 a call discriminator which selectively enables at least one of the voice exchange
and the data exchange.

10 27. The signal processing system of claim 26 wherein the call discriminator is capable
of discriminating between the voice signals and the data signals, the voice exchange being
enabled for the voice signals and the data exchange being enabled for the data signals.

15 28. The signal processing system of claim 26 wherein the data signals from the second
telephony device are modulated by a voiceband carrier, and the data exchange comprises a data
pump capable of demodulating the data signals from the second telephony device for
transmission on the packet based network and remodulating the data signals from the packet
based network with the voiceband carrier for transmission to the second telephony device.

20 29. The signal processing system of claim 28 wherein the data exchange comprises
a jitter buffer capable of receiving packets of the data signals of varying delay from the packet
based network and compensating for the delay variation of the data signal packets.

25 30. The signal processing system of claim 29 wherein the jitter buffer outputs an
isochronous stream of the received data signals.

30 31. The signal processing system of claim 29 wherein the data pump is capable of
transmitting the received data signals to the second telephony device at a transmit rate:

35 32. The signal processing system of claim 31 wherein the jitter buffer compensates
for the delay variation of the data signal packets by holding a number of the received data signals,
and wherein the data exchange further comprises a clock synchronizer which adaptively adjusts
the transmit rate of the data pump in response the number of the received data signals in the jitter
buffer.

 33. The signal processing system of claim 31 wherein the jitter buffer compensates
for the delay variation of the data signal packets by holding a number of the received data signals,
and wherein the data exchange further comprises spoof logic which provides spoof data to the
data pump when the number of the received data signals held in the jitter buffer is below a
threshold.

1 34. The signal processing system of claim 26 wherein the voice exchange comprises
a jitter buffer capable of receiving packets of the voice signals of varying delay from the packet
based network and compensating for the delay variation of the voice signal packets.

5 35. The signal processing system of claim 34 wherein the jitter buffer outputs an
isochronous stream of the received voice signals.

 36. The signal processing system of claim 34 wherein the jitter buffer comprises a
voice queue which buffers the received voice signals for a holding time, and a voice synchronizer
which adaptively adjusts the holding time of the voice queue.

10 37. The signal processing system of claim 36 further comprising a tone exchange
capable of exchanging DTMF signals between the first telephony device and the packet based
network, the DTMF exchange comprising a DTMF queue capable of buffering packets of the
DTMF signals from the packet based network, and a tone generator which generates a DTMF
15 tone responsive to the buffered DMTF signals, the DMTF queue outputting a signal to the voice
synchronizer to suppress the buffered voice signals when the DTMF signals are in the DTMF
queue.

 38. The signal processing system of claim 26 wherein the voice exchange comprises
a voice decoder capable of decoding packets of the voice signals from the packet based network
20 for transmission to the first telephony device, a voice activity detector which detects the voice
signals without speech, and a comfort noise generator which inserts comfort noise in place of the
voice signals without speech.

 39. The signal processing system of claim 38 wherein the voice exchange further
25 comprises a comfort noise estimator which generates comfort noise parameters from at least a
portion of the voice signals without speech, the comfort noise generator being responsive to the
comfort noise parameters.

 40. The signal processing system of claim 26 wherein the voice exchange comprises
a voice decoder capable of decoding packets of the voice signals from the packet based network
30 for transmission to the first telephony device, a voice activity detector which detects lost voice
signals, and a lost packet recovery engine which processes the voice signals to compensate for
the lost voice signals.

 41. The signal processing system of claim 26 wherein the voice exchange comprises
35 a voice encoder capable of encoding the voice signals from the first telephony device for
transmission on the packet based network, and a voice activity detector which suppresses the
voice signals without speech.

1 42. The signal processing system of claim 41 wherein the voice exchange further comprises a comfort noise estimator which generates comfort noise parameters when the voice activity detector suppresses the voice signals without speech.

5 43. The signal processing system of claim 26 wherein the voice exchange further comprises a decoder capable of decoding packets of the voice signals from the packet based network, and an echo canceller capable of cancelling decoded voice signal echos on incoming voice signals from the first telephony device.

10 44. The signal processing system of claim 43 wherein the voice exchange further comprises a non-linear processor which mutes the incoming voice signals when the incoming voice signals do not comprise speech and the echo canceller detects the decoded voice signals with speech.

15 45. The signal processing system of claim 26 wherein the voice exchange comprises a voice encoder capable of encoding the voice signals from the first telephony device into voice signal packets for the packet based network.

20 46. The signal processing system of claim 45 further comprising a tone exchange comprising a DTMF detector capable of detecting a DTMF signal from the first telephony device and generating a DTMF packet for the packet based network in response to the DTMF signal, the DTMF detector muting the voice signal packets when a DTMF signal is detected.

25 47. The signal processing system of claim 26 further comprising a fax exchange capable of exchanging fax signals from a third telephony device with demodulated fax signals from the packet based network, wherein the call discriminator selectively enables the fax exchange.

30 48. The signal processing system of claim 47 wherein the fax signals from the third telephony device are modulated by a voiceband carrier, and the fax exchange comprises a data pump capable of demodulating the fax signals from the third telephony device for transmission on the packet based network, and remodulating the demodulated fax signals from the packet based network with the voiceband carrier for transmission to the third telephony.

35 49. A method of processing signals, comprising:
 exchanging voice signals between a network line and a packet based network; and
 simultaneously exchanging data signals from the network line with demodulated data signals from the packet based network.

 50. The method of claim 49 further comprising discriminating between the voice signals and the data signals from the network line, and selectively invoking at least one of the voice signal exchange and the data signal exchange based on said discrimination.

1 51. The method of claim 49 wherein the data signals from the network line are
modulated by a voiceband carrier, and the data exchange comprises demodulating the data signals
from the network line for transmission on the packet based network and remodulating the data
signals from the packet based network with the voiceband carrier for transmission on the network
5 line.

 52. The method of claim 49 wherein the voice exchange further comprises receiving
packets of the signals of varying delay from the packet based network, and compensating for the
delay variation of the signal packets.

10 53. The method of claim 52 wherein the signal packet compensation comprises
generating an isochronous stream of the received signals.

 54. The method of claim 52 wherein the signal packet compensation comprises
adaptively buffering the received signals.

15 55. The method of claim 49 wherein the voice signal exchange comprises receiving
packets of the voice signals from the packet based network, identifying the received voice signals
without speech, and inserting comfort noise in place of the identified voice signals without
speech.

20 56. The method of claim 55 wherein the comfort noise insertion comprises estimating
comfort noise in response to at least a portion of the received voice signals without speech.

 57. The method of claim 49 wherein the voice signal exchange comprises receiving
packets of the voice signals from the packet based network, detecting lost voice signals, decoding
the received voice signals for transmission to the network line, and processing the decoded voice
25 signals to compensate for the lost voice signals.

 58. The method of claim 49 further comprising exchanging DTMF signals between
the network line and the packet based network.

30 59. The method of claim 58 wherein the DTMF signal exchange comprises receiving
packets of the DTMF signals from the packet based network, and generating at least one DTMF
tone from the DTMF signals.

 60. The method of claim 59 wherein the voice signal exchange comprises receiving
35 packets of the voice signals from the packet based network, and the DTMF signal exchange
further comprises muting the received voice signals when the DTMF signal packets are received.

1 61. The method of claim 49 wherein the voice signal exchange comprises decoding
packets of the voice signals from the packet based network, receiving voice signals from the
network line and canceling decoded voice signal echos on the received voice signals.

5 62. The method of claim 49 wherein the voice signal exchange comprises encoding
the voice signals from the network line into voice signal packets for transmission on the packet
based network.

10 63. The method of claim 62 further comprising exchanging DTMF signals between
the network line and the packet based network, wherein the DTMF signal exchange comprises
detecting DTMF signals from the network line, generating DTMF signal packets for the packet
based network in response to the DTMF signals, and muting the voice signal packets when the
DTMF signals are detected..

15 64. The method of claim 49 wherein the voice signal exchange comprises receiving
the voice signals from the network line and suppressing the received voice signals when the
received voice signals do not comprise speech.

 65. The method of claim 64 wherein the suppression of the received voice signals
comprises generating comfort noise parameters in place thereof.

20 66. The method of claim 49 further comprising exchanging fax signals from the
network line with demodulated fax signals from the packet based network

25 67. The method of claim 66 wherein the fax signals from the network line are
modulated by a voiceband carrier, and the fax exchange comprises demodulating the fax signals
from the network line for transmission on the packet based network and remodulating the fax
signals from the packet based network with the voiceband carrier for transmission on the network
line.

30 68. The method of claim 67 wherein the signal discrimination further comprises
discriminating the fax signals from the network line, and selectively invoking the fax exchange
based on said discrimination.

35 69. The method of claim 49 wherein the data signal exchange further comprises
receiving packets of the data signals from the packet based network, holding a number of the
received data signals in a buffer, and generating spoof data when the number of the data signals
in the buffer is below a threshold.

 70. The method of claim 49 wherein the data signal exchange further comprises
receiving packets of the data signals from the packet based network, holding a number of the
received data signals in a buffer, transmitting the buffered data signals to the network line at a

1 transmit rate, and adaptively adjusting the transmit rate in response the number of the received data signals in the buffer.

5 71. The method of claim 49 wherein the data signal exchange further comprises receiving data rate codes from the packet based network, and setting a data rate of a telephony device coupled to the network line in response to the received data rate codes.

72. The method of claim 49 wherein the network line comprises a circuit switched network line.

10 73. The method of claim 72 wherein the circuit switched network line comprises a public switching telephone network line.

15 74. A method of processing signals, comprising:
exchanging voice signals between a first telephony device and a packet based network;
simultaneously exchanging data signals from a second telephony device with demodulated data signals from the packet based network; and
discriminating between the voice signals and the data signals, and invoking at least one of the voice exchange and the data exchange based on said discrimination.

20 75. The method of claim 74 wherein the data signals from the second telephony device are modulated by a voiceband carrier, and the data exchange comprises demodulating the data signals from the second telephony device for transmission on the packet based network and remodulating the data signals from the packet based network with the voiceband carrier for transmission to the second telephony device.

25 76. The method of claim 74 further comprising receiving packets of the signals of varying delay from the packet based network, and compensating for the delay variation of the signal packets.

30 77. The method of claim 76 wherein the signal packet compensation comprises generating an isochronous stream of the received signals.

78. The method of claim 76 wherein the signal packet compensation comprises adaptively buffering the received signals.

35 79. The method of claim 74 wherein the voice signal exchange comprises receiving packets of the voice signals from the packet based network, identifying the received voice signals without speech, and inserting comfort noise in place of the identified voice signals without speech.

1 80. The method of claim 79 wherein the comfort noise insertion comprises estimating
comfort noise in response to at least a portion of the received voice signals without speech.

5 81. The method of claim 74 wherein the voice signal exchange comprises receiving
packets of the voice signals from the packet based network, detecting lost voice signals, decoding
the received voice signals for transmission to the first telephony device, and processing the
decoded voice signals to compensate for the lost voice signals.

10 82. The method of claim 74 wherein the signal discrimination further comprises
discriminating between the voice signals, the data signals, and DTMF signals, and further
comprising exchanging the discriminated DTMF signals between the first telephony device and
the packet based network.

15 83. The method of claim 82 wherein the DTMF signal exchange comprises receiving
packets of the DTMF signals from the packet based network, and generating at least one DTMF
tone from the DTMF signals.

 84. The method of claim 83 wherein the voice signal exchange comprises receiving
packets of the voice signals from the packet based network, and the DTMF signal exchange
further comprises muting the received voice signals when the DTMF signal packets are received.

20 85. The method of claim 74 wherein the voice signal exchange comprises decoding
packets of the voice signals from the packet based network, receiving voice signals from the first
telephony device and canceling decoded voice signal echos on the received voice signals.

25 86. The method of claim 74 wherein the voice signal exchange comprises encoding
the voice signals from the first telephony device into voice signal packets for transmission on the
packet based network.

30 87. The method of claim 86 further comprising exchanging the DTMF signals
between the first telephony device and the packet based network, wherein the DTMF signal
exchange comprises detecting DTMF signals from the first telephony device, generating DTMF
signal packets for the packet based network in response to the DTMF signals, and muting the
voice signal packets when the DTMF signals are detected.

35 88. The method of claim 74 wherein the voice signal exchange comprises receiving
the voice signals from the first telephony device and suppressing the received voice signals when
the received voice signals do not comprise speech.

 89. The method of claim 88 wherein the suppression of the received voice signals
comprises generating comfort noise parameters in place thereof.

1 90. The method of claim 74 wherein further comprising exchanging fax signals from
a third telephony device with demodulated fax signals from the packet based network, wherein
the signal discrimination comprises selectively invoking the fax exchange.

5 91. The method of claim 90 wherein the fax signals from the third telephony device
are modulated by a voiceband carrier, and the fax exchange comprises a data pump capable of
demodulating the fax signals from the third telephony device for transmission on the packet
based network, and remodulating the fax signals from the packet based network with the
voiceband carrier for transmission to the third telephony device.

10 92. The method of claim 74 wherein the data signal exchange further comprises
receiving packets of the data signals from the packet based network, holding a number of the
received data signals in a buffer, and spoofing the second telephony device when the number of
the data signals in the buffer is below a threshold.

15 93. The method of claim 74 wherein the data signal exchange further comprises
receiving packets of the data signals from the packet based network, holding a number of the
received data signals in a buffer, transmitting the buffered data signals to the second telephony
device at a transmit rate, and adaptively adjusting the transmit rate in response the number of the
received data signals in the buffer.

20 94. The method of claim 74 wherein the data signal exchange further comprises
receiving data rate codes from the packet based network, and setting a data rate of the second
telephony device in response to the received data rate codes.

25 95. A signal transmission system, comprising:
a first telephony device which transmits and receives voice signals;
a second telephony device different from the first telephony device;
a packet based network; and
a signal processing system coupling the first and the second telephony devices to the
packet based network, the signal processing system comprising a full duplex data exchange
which exchanges data signals from the second telephony device with demodulated data signals
30 from the packet based network.

 96. The signal transmission system of claim 95 further comprising a circuit switched
network coupling the first and the second telephony devices to the signal processing system.

35 97. The signal transmission system of claim 96 wherein the circuit switched network
comprises a public switching telephone network.

 98. The signal transmission system of claim 95 wherein the packet based network
comprises internet protocol.

1 99. The signal transmission system of claim 95 wherein the packet based network comprises frame relay.

5 100. The signal transmission system of claim 95 wherein the packet based network comprises asynchronous transfer mode.

 101. The signal transmission system of claim 95 wherein the packet based network comprises a time division multiplexing network.

10 102. The signal transmission system of claim 95 wherein the data signals from the second telephony are modulated by a voiceband carrier, and the data exchange comprises a data pump which demodulates the data signals from the second telephony device for transmission on the packet based network and remodulates the data signals from the packet based network with the voiceband carrier for transmission to the second telephony device.

15 103. The signal transmission system of claim 95 wherein the data exchange comprises a jitter buffer which receives packets of the data signals of varying delay from the packet based network and compensates for the delay variation of the data signal packets.

20 104. The signal transmission system of claim 103 wherein the jitter buffer outputs an isochronous stream of the received data signals.

 105. The signal transmission system of claim 103 wherein the data pump transmits the received data signals to the second telephony device at a transmit rate.

25 106. The signal transmission system of claim 105 wherein the jitter buffer compensates for the delay variation of the data signal packets by holding a number of the received data signals, and wherein the data exchange further comprises a clock synchronizer which adaptively adjusts the transmit rate of the data pump in response the number of the received data signals in the jitter buffer.

30 107. The signal transmission system of claim 103 wherein the jitter buffer compensates for the delay variation of the data signal packets by holding a number of the received data signals, and wherein the data exchange further comprises spoof logic which provides spoof data to the data pump when the number of the received data signals held in the jitter buffer is below a threshold.

35 108. The signal transmission system of claim 95 wherein the signal processing system further comprises a voice exchange which exchanges the voice signals between the first telephony device and the packet based network.

1 109. The signal transmission system of claim 108 wherein the signal processing system further comprises a call discriminator which discriminates between a first telephony device transmission and a second telephony device transmission, the call discriminator invoking at least one of the voice exchange and the data exchange based on said discrimination.

5 110. The signal transmission system of claim 108 wherein the voice exchange comprises a jitter buffer which receives packets of the voice signals of varying delay from the packet based network and compensates for the delay variation of the voice signal packets.

10 111. The signal transmission system of claim 110 wherein the jitter buffer outputs an isochronous stream of the received voice signals.

 112. The signal transmission system of claim 110 wherein the jitter buffer comprises a voice queue which buffers the received voice signals for a holding time, and a voice synchronizer which adaptively adjusts the holding time of the voice queue.

15 113. The signal transmission system of claim 112 wherein the signal processing system further comprises a tone exchange which exchanges DTMF signals between the first telephony device and the packet based network, the DTMF exchange comprising a DTMF queue which buffers packets of the DTMF signals from the packet based network, and a tone generator which generates a DTMF tone responsive to the buffered DMTF signals, the DMTF queue outputting a signal to the voice synchronizer to suppress the buffered voice signals when the DTMF signals are in the DTMF queue.

20 114. The signal processing system of claim 108 wherein the voice exchange comprises a voice decoder which decodes packets of the voice signals from the packet based network for transmission to the first telephony device, a voice activity detector which detects the voice signals without speech, and a comfort noise generator which inserts comfort noise in place of the voice signals without speech.

25 115. The signal processing system of claim 114 wherein the voice exchange further comprises a comfort noise estimator which generates comfort noise parameters from at least a portion of the voice signals without speech, the comfort noise generator being responsive to the comfort noise parameters.

30 116. The signal processing system of claim 108 wherein the voice exchange comprises a voice decoder which decodes packets of the voice signals from the packet based network for transmission to the first telephony device, a voice activity detector which detects lost voice signals, and a lost packet recovery engine which processes the voice signals to compensate for the lost voice signals.

1 117. The signal processing system of claim 108 wherein the voice exchange comprises
a voice encoder which encodes voice signals from the first telephony device for transmission on
the packet based network, and a voice activity detector which suppresses the voice signals
without speech.

5 118. The signal processing system of claim 117 further comprising a comfort noise
estimator which generates comfort noise parameters when the voice activity detector suppresses
the voice signals without speech.

10 119. The signal processing system of claim 108 wherein the voice exchange further
comprises a decoder which decodes packets of the voice signals from the packet based network,
and an echo canceller which cancels decoded voice signal echos on incoming voice signals from
the first telephony device.

15 120. The signal processing system of claim 119 further comprising a non-linear
processor which mutes the incoming voice signals when the incoming voice signals do not
comprise speech and the echo canceller detects the decoded voice signals with speech.

20 121. The signal processing system of claim 108 wherein the voice exchange comprises
a voice encoder which encodes the voice signals from the first telephony device into voice signal
packets for the packet based network.

25 122. The signal processing system of claim 121 further comprising a tone exchange
comprising a DTMF detector which detects a DTMF signal from the first telephony device and
generates a DTMF packet for the packet based network in response to the DTMF signal, the
DTMF detector muting the voice signal packets when the DTMF signal is detected.

30 123. The signal transmission system of claim 95 further comprising a third telephony
device coupled to the signal processing system.

35 124. The signal transmission system of claim 123 wherein the third telephony
comprises a fax.

 125. The signal processing system of claim 124 further comprising a fax exchange
which exchanges fax signals from the third telephony device with demodulated data signals from
the packet based network.

 126. The signal processing system of claim 125 wherein the fax signals from the third
telephony device are modulated by a voiceband carrier, and the fax exchange comprises a data
pump which demodulates the fax signals from the third telephony device for transmission on the
packet based network and remodulates the fax signals from the packet based network with the
voiceband carrier for transmission to the third telephony device.

1 127. The signal transmission system of claim 95 wherein the first telephony device comprises a telephone.

5 128. The signal transmission system of claim 95 wherein the second telephony device comprises a modem.

10 129. A method of transmitting data, comprising:
 negotiating a data rate between a rate negotiator and a first telephony device; and
 renegotiating the negotiated data rate between the rate negotiator and a system
 having a second telephony device to allow data transmission between the first and second
 telephony devices.

 130. The method of claim 129 wherein the first and second telephony devices each comprises a modem.

15 131. The method of claim 129 wherein the data rate negotiation comprises setting the negotiated data rate based on a first telephony device data rate and a rate negotiator data rate.

 132. The method of claim 131 wherein the negotiated data rate is set to the lower of the first telephony device data rate and the rate negotiator data rate.

20 133. The method of claim 129 wherein the data rate renegotiation comprises setting the renegotiated data rate based on a system data rate and the negotiated data rate.

25 134. The method of claim 133 wherein the system further comprises a data exchange, the method further comprising negotiating a second telephony device data rate between the data exchange and the second telephony device, and setting the system data rate based on the negotiated second telephony device data rate.

 135. The method of claim 133 wherein the renegotiated data rate is set to the lower of the system data rate and the negotiated data rate.

30 136. The method of claim 133 wherein the data rate renegotiation is performed over a packet based network.

 137. The method of claim 136 wherein the data rate renegotiation further comprises inhibiting receipt of data packets from the packet based network.

35 138. The method of claim 133 wherein the data rate renegotiation further comprises resetting the first telephony device with the renegotiated data rate.

1 139. The method of claim 138 wherein the first telephony device is reset by retraining
the first telephony device with the renegotiated data rate.

5 140. A method of synchronizing a data rate, comprising:
initializing a data rate;
receiving a data rate from a first telephony device;
setting a negotiated data rate based on the initial data rate and the data rate for the
first telephony device;
receiving a data rate from a system; and
10 setting a renegotiated data rate based on the negotiated data rate and the system
data rate.

 141. The method of claim 140 wherein the negotiated data rate is set to the lower of
the initial data rate and the data rate for the first telephony device.

15 142. The method of claim 140 wherein the setting of the negotiated data rate comprises
setting the data rate for the first telephony device to the negotiated data rate.

 143. The method of claim 142 wherein the setting of the renegotiated data rate
comprises resetting the set data rate for the first telephony device to the renegotiated data rate.

20 144. The method of claim 140 wherein the renegotiated data rate is set to the lower of
the negotiated data rate and the system data rate.

25 145. The method of claim 140 wherein the system comprises a second telephony device
and a data exchange, and wherein the receiving of a second data rate comprises negotiating the
system data rate between the second telephony device and the [a] data exchange.

 146. The method of claim 145 wherein the system data rate negotiation comprises
setting a data rate for the second telephony device to the system data rate.

30 147. The method of claim 146 wherein the setting of the renegotiated data rate
comprises resetting the set data rate for the second telephony device to the renegotiated data rate.

 148. The method of claim 145 wherein the second telephony device comprises a
modem.

35 149. The method of claim 140 wherein the first telephony device comprises a modem.

 150. The method of claim 140 wherein the renegotiation of the data rate is over a
packet based network.

1 151. The method of claim 150 wherein the data rate renegotiation comprising inhibiting receipt of data packets.

5 152. A method of synchronizing a data rate, comprising:
 exchanging data rates between a first data exchange and a first telephony device;
 negotiating a first data rate based on the exchanged data rates between the first data exchange and the first telephony device;
 exchanging data rates between a second data exchange and a second telephony device;
 negotiating a second data rate based on the exchanged rates between the second data exchange and the second telephony device;
10 exchanging the first and the second data rates over a packet based network; and
 negotiating a third data rate based on the exchanged first and second data rates.

15 153. The method of claim 152 further comprising resetting the first and the second telephony devices with the third data rate.

 154. The method of claim 153 wherein each of the first and second telephony devices are reset by retraining each of the first and second telephony devices with the third data rate.

20 155. The method of claim 152 wherein the first and the second telephony devices each comprises a modem.

 156. The method of claim 152 wherein the first data rate negotiation comprises setting the first data rate to the lower of the exchanged data rates between the first data exchange and the first telephony device.

25 157. The method of claim 152 wherein the second data rate negotiation comprises setting the second data rate to the lower of the exchanged data rates between the second data exchange and the second telephony device.

30 158. The method of claim 152 wherein the third data rate negotiation comprises setting the third data rate to the lower of the exchanged first and second data rates.

 159. The method of claim 152 wherein the third data rate negotiation comprises inhibiting receipt of data packets at each of the first and second data exchanges.

35 160. A data exchange comprising a rate negotiator capable of negotiating a data rate with a first telephony device, and renegotiating the negotiated data rate with a system comprising a second telephony device to allow data transmission between the first and second telephony devices.

1 161. The data exchange of claim 160 wherein the rate negotiator initializes an initial data rate, the negotiated data rate being based on an initial data rate of the first telephony device and the initial data rate of the rate negotiator.

5 162. The data exchange of claim 160 wherein the negotiated data rate comprises the lower of the initial first telephony device data rate and the initial rate negotiator data rate.

 163. The data exchange of claim 160 wherein the renegotiated data is based on a data rate for the system and the negotiated data rate.

10 164. The data exchange of claim 163 wherein the renegotiated data rate comprises the lower of the system data rate and the negotiated data rate.

 165. The data exchange of claim 163 wherein the rate negotiator is further capable of resetting the first telephony device with the renegotiated data rate.

15 166. The data exchange of claim 165 wherein the rate negotiator rests the first telephony device by retraining the first telephony device with the renegotiated data rate.

20 167. The data exchange of claim 160 further comprising a data pump capable of exchanging data signals between a circuit switched network and a packet based network at the renegotiated data rate.

 168. The data exchange of claim 167 wherein the rate negotiator inhibits receipt of the data signals from the packet based network during data rate renegotiation.

25 169. A signal transmission system, comprising:
 a first telephony device having a data rate;
 a first data exchange having a data rate;
 a first rate negotiator which exchanges the data rates between the first data exchange and the first telephony device and negotiates a first data rate based on the exchanged data rates between the first data exchange and the first telephony device;
30 a second telephony device having a data rate;
 a second data exchange having a data rate;
 a second rate negotiator which exchanges the data rates between the second data exchange and the second telephony device and negotiates a second data rate based on the exchanged data rates between the second data exchange and the second telephony device,
35 wherein the first and the second rate negotiators cooperate to exchange the first and the second data rates and negotiate a third data rate based on the exchanged first and second data rates; and
 a packet based network coupling the first data exchange to the second data exchange.

1 170. The signal transmission system of claim 169 wherein each of the first and the second rate negotiators sets the data rate of its respective telephony device to the third data rate.

5 171. The signal transmission system of claim 169 wherein the first and the second telephony devices each comprises a modem.

 172. The signal transmission system of claim 169 wherein the first data rate comprises the lower of the exchanged data rates between the first data exchange and the first telephony device.

10 173. The signal transmission system of claim 169 wherein the second data rate comprises the lower of the exchanged data rates between the second data exchange and the second telephony device.

15 174. The signal transmission system of claim 169 wherein the third data rate comprises the lower of the first and the second data rates.

 175. The signal transmission system of claim 169 wherein each of the first and the second rate negotiators inhibit reception of data packets from the packet based network during the third data rate negotiation.

20 176. The signal transmission system of claim 169 wherein each of the first and the second data exchanges comprises a data pump which exchanges data signals between the packet based network and its respective telephony device.

25 177. A method of compensating a signal, comprising:
 averaging the signal in a first phase; and
 combining the average signal with the signal in a second phase.

 178. The method of claim 177 wherein the signal averaging comprises accumulating the signal and scaling the accumulated signal by a constant.

30 179. The method of claim 178 wherein the accumulation of the signal comprises accumulating the signal a number of times, and wherein the constant is a function of the number of times the signal is accumulated.

35 180. The method of claim 179 wherein the constant is inversely proportional to the number of times the signal is accumulated.

 181. The method of claim 177 wherein the signal averaging comprises accumulating the signal a number of times, and scaling the accumulated signal by a constant inversely proportional to the number of times the signal is accumulated, and wherein the combining of the

1 average signal with the signal comprises incrementally combining the average signal with the
signal over time, the method further comprising scaling the signal prior to combining it with the
average signal.

5 182. The method of claim 177 further comprising scaling the signal prior to combining
it with the average signal.

183. The method of claim 177 wherein the combining of the average signal with the
signal comprises incrementally combining the average signal with the signal over time.

10 184. The method of claim 183 wherein the combining of the average signal with the
signal further comprises combining the average signal with the signal in incremental linear steps
over time.

15 185. A method of timing recovery, comprising:
receiving a signal having a data rate;
sampling the signal at a sampling rate;
estimating a timing error between the data rate and the sampling rate;
averaging the estimated timing error during a first phase;
combining the averaged estimated timing error with the estimated timing error
during a second phase; and
20 adjusting the sampling rate as a function of the combined averaged estimated
timing error and the estimated timing error.

25 186. The method of claim 185 wherein the averaging of the estimated timing error
comprises accumulating the estimated timing error and scaling the accumulated estimated timing
error by a constant.

187. The method of claim 186 wherein the estimated timing error accumulation
comprises accumulating the estimated timing error a number of times, and wherein the constant
is a function of the number of times the estimated timing error is accumulated.

30 188. The method of claim 187 wherein the constant is inversely proportional to the
number of times the estimated timing error is accumulated.

35 189. The method of claim 185 wherein the averaging of the estimated timing error
comprises accumulating the estimated timing error a number of times, and scaling the
accumulated estimated timing error by a constant inversely proportional to the number of times
the estimated timing error is accumulated, and wherein the combining of the average estimated
timing error with the estimated timing error comprises incrementally combining the average
estimated timing error with the estimated timing error over time, the method further comprising
scaling the estimated timing error prior to combining it with the average estimated timing error.

1 190. The method of claim 185 further comprising scaling the estimated timing error before combining it with the average estimated timing error.

5 191. The method of claim 185 wherein the combining of the average estimated timing error with the estimated timing error comprises incrementally combining the average estimated timing error with the estimated timing error over time.

10 192. The method of claim 191 wherein the combining of the average estimated timing error with the estimated timing error further comprises combining the average estimated timing error with the estimated timing error in incremental linear steps over time.

15 193. A method of timing recovery, comprising:
receiving a signal at a data rate;
sampling the signal at a sampling rate;
measuring a phase error between the data rate and the sampling rate;
estimating a frequency error between the data rate and the sampling rate during
a first phase;
combining the phase error and the frequency error during a second phase; and
adjusting the sampling rate as a function of the combined frequency error and
phase error.

20 194. The method of claim 193 wherein the frequency error estimation comprises averaging the measured phase error over time.

25 195. The method of claim 193 wherein the frequency error estimation comprises accumulating the measured phase error and scaling the accumulated measured phase error by a constant.

 196. The method of claim 195 wherein the measured phase error accumulation comprises accumulating the measured phase error a number of times, and wherein the constant is a function of the number of times the measured phase error is accumulated.

30 197. The method of claim 196 wherein the constant is inversely proportional to the number of times the measured phase error is accumulated.

35 198. The method of claim 193 wherein the frequency error estimation comprises accumulating the measured phase error a number of times, and scaling the accumulated measured phase error by a constant inversely proportional to the number of times the measured phase error is accumulated, and wherein the combining of the estimated frequency error with the measured phase error comprises incrementally combining the estimated frequency error with the measured phase error over time, the method further comprising scaling the measured phase error prior to combining it with the average estimated timing error.

1 199. The method of claim 193 further comprising scaling the phase error before combining it with the frequency error.

5 200. The method of claims 199 wherein the combining of the estimated frequency error with the measured phase error comprises combining the frequency error with the phase error in incremental linear steps over time.

10 201. The method of claim 200 wherein the combining of the estimated frequency error with the measured phase error further comprises combining the frequency error with the phase error in incremental linear steps over time.

15 202. A signal compensator, comprising:
 an estimator which estimates an average of the signal in a first phase; and
 a combiner which combines the signal with the estimated average signal in a second phase.

20 203. The signal compensator of claim 202 wherein the estimator comprises an accumulator which accumulates the signal a number of times, and a first multiplier which scales the accumulated signal by a constant inversely proportional to the number of times the signal is accumulated, and wherein the combiner incrementally combines the estimated average signal with the signal over time, the signal compensator further comprising a second multiplier which scales the signal prior to combining it with the estimated average signal.

25 204. The signal compensator of claim 202 wherein the estimator comprises an accumulator which accumulates the signal, and a multiplier which scales the accumulated signal.

30 205. The signal compensator of claim 204 wherein the accumulator accumulates the signal a number of times, and the multiplier scales the accumulated signal by a constant which is a function of the number of times the signal is accumulated.

35 206. The signal compensator of claim 205 wherein the constant is inversely proportional to the number of times the signal is accumulated.

 207. The signal compensator of claim 202 further comprising a multiplier which scales the signal prior to combining it with the estimated average signal.

 208. The signal compensator of claim 202 wherein the combiner incrementally combines the estimated average signal with the signal over time.

 209. The signal compensator of claim 208 wherein the combiner incrementally combines the estimated average signal with the signal in incremental linear steps over time.

1 210. A timing recovery system, comprising:
a sampler capable of sampling a signal at a sampling rate, the signal having a data
rate;
a timing error estimator which estimates a phase error between the data rate and
5 the sampling rate;
a frequency offset estimator which estimates a frequency error between the data
rate and the sampling rate during a first phase;
a combiner which combines the estimated phase error and the estimated frequency
error during a second phase; and
a clock adjuster which adjusts the sampling rate of the sampler as a function of
10 the combined estimated frequency error and estimated phase error.

211. The timing recovery system of claim 210 further comprising a multiplier which
scales the estimated phase error before the estimated phase error is combined with the estimated
frequency error by the combiner.

15 212. The timing recovery system of claim 210 wherein the frequency offset estimator
comprises an accumulator which accumulates the estimated phase error, and a multiplier which
scales the accumulated estimated phase error to generate the estimated frequency error.

20 213. The timing recovery system of claim 212 wherein the accumulator accumulates
the estimated phase error a number of times, and the multiplier scales the accumulated estimated
phase error by a constant which is a function of the number of times the estimated phase error
is accumulated.

25 214. The timing recovery system of claim 213 wherein the constant is inversely
proportional to the number of times the estimated timing error is accumulated.

30 215. The timing recovery system of claim 212 wherein the frequency offset estimator
comprises an accumulator which accumulates the estimated phase error a number of times, and
a first multiplier which scales the accumulated estimated phase error by a constant inversely
proportional to the number of times the estimated phase error is accumulated to generate the
estimated frequency error, and wherein the combiner incrementally combines the estimated
frequency error with the estimated phase error over time, the timing recovery system further
comprising a second multiplier which scales the estimated phase error before it is combined with
the estimated frequency error by the combiner.

35 216. The timing recovery system of claim 210 wherein the combiner incrementally
combines the estimated frequency error with the estimated phase error over time.

217. The timing recovery system of claim 216 wherein the combiner combines the
estimated frequency error with the estimated phase error in incremental linear steps over time.

1 218. A data transmission system, comprising:
a telephony device which outputs a signal having a data rate; and
a data exchange coupled to the telephony device, the data exchange having a
sampler capable of sampling the signal at a sampling rate, a timing error estimator which
5 estimates a phase error between the data rate and the sampling rate, a frequency offset estimator
which estimates a frequency error between the data rate and the sampling rate during a first
phase, a combiner which combines the estimated phase error and the estimated frequency error
during a second phase, and a clock adjuster which adjusts the sampling rate of the sampler as a
function of the combined estimated frequency error and estimated phase error.

10 219. The data transmission system of claim 218 wherein the accumulator accumulates
the estimated phase error a number of times, and the multiplier scales the accumulated estimated
phase error by a constant which is a function of the number of times the estimated phase error
is accumulated.

15 220. The data transmission system of claim 219 wherein the constant is inversely
proportional to the number of times the estimated timing error is accumulated.

20 221. The data transmission system of claim 218 wherein the frequency offset estimator
comprises an accumulator which accumulates the estimated phase error a number of times, and
a first multiplier which scales the accumulated estimated phase error by a constant inversely
proportional to the number of times the estimated phase error is accumulated to generate the
estimated frequency error, and wherein the combiner incrementally combines the estimated
frequency error with the estimated phase error over time, the data exchange further comprising
a second multiplier which scales the estimated phase error before it is combined with the
estimated frequency error by the combiner.

25 222. The data transmission system of claim 218 wherein the data exchange further
comprises a multiplier which scales the estimated phase error before the estimated phase error
is combined with the estimated frequency error by the combiner.

30 223. The data transmission system of claim 218 wherein the frequency offset estimator
comprises an accumulator which accumulates the estimated phase error, and a multiplier which
scales the accumulated estimated phase error to generate the estimated frequency error.

 224. The data transmission system of claim 218 wherein the combiner incrementally
combines the estimated frequency error with the estimated phase error over time.

35 225. The data transmission system of claim 224 wherein the combiner combines the
estimated frequency error with the estimated phase error in incremental linear steps over time.

1 226. The data transmission system of claim 218 wherein the telephony device comprises a modem

5 227. The data transmission system of claim 218 further comprising a PSTN line coupling the telephony device to the data exchange.

10 228. A method for replicating locally a signal generated remotely, comprising:
 estimating a first parameter of the signal remotely;
 estimating a second parameter of the signal locally, the second parameter being different from the first parameter; and
 modifying a second signal to replicate the signal as a function of the estimated first and second parameters.

15 229. The method of claim 228 wherein the second signal comprises white noise.

20 230. The method of claim 228 wherein the first parameter comprises energy of the signal and the second parameter comprises spectral characteristics of the signal.

25 231. The method of claim 230 wherein the modification of the second signal comprises scaling the second signal as a function of the estimated energy, and filtering the second signal as a function of the estimated spectral characteristics.

30 232. The method of claim 231 wherein the second signal comprises white noise.

35 233. The method of claim 231 wherein the estimation of the spectral characteristics comprises estimating filter coefficients which model the spectral shape of the signal, the filtering of the second signal being a function of the estimated filter coefficients.

 234. The method of claim 233 wherein the estimation of the filter coefficients comprises calculating autocorrelation coefficients, the filter coefficients being a function of the autocorrelation coefficients.

 235. The method of claim 228 wherein the estimation of the filter coefficients comprises calculating linear prediction coefficients, the filter coefficients being a function of the linear prediction coefficients.

 236. A method for replicating, at a near end, far end background noise of a signal generated by a far end, comprising:
 estimating a first parameter of the far end background noise at the far end;
 transmitting the first parameter and the signal from the far end to the near end;
 estimating a second parameter different from the first parameter of the far end background noise at the near end; and

1 modifying a noise signal to replicate the far end background noise as a function
of the estimated first and second parameters.

237. The method of claim 236 wherein the noise signal comprises white noise.

5 238. The method of claim 236 wherein the first parameter comprises energy of the far
end background noise and the second parameter comprises spectral characteristics of the far end
background noise.

10 239. The method of claim 238 wherein the modification of the noise signal comprises
scaling the noise signal as a function of the estimated energy, and filtering the noise signal as a
function of the estimated spectral characteristics.

15 240. The method of claim 239 wherein the estimation of the spectral characteristics
comprises estimating filter coefficients which model the spectral shape of the far end background
noise, the filtering of the noise signal being a function of the estimated filter coefficients.

241. The method of claim 240 wherein the estimation of the filter coefficients
comprises calculating autocorrelation coefficients, the filter coefficients being a function of the
autocorrelation coefficients.

20 242. The method of claim 241 wherein the estimation of the filter coefficients
comprises calculating linear prediction coefficients, the filter coefficients being a function of the
linear prediction coefficients.

25 243. A local receiver for replicating a signal generated by a remote transmitter, the
local receiver adapted to receive the signal and a first parameter of the signal, the local receiver,
comprising:

 a signal estimator to estimate a second parameter of the signal different from the
first parameter; and

 a signal generator to modify a second signal to replicate the signal as a function
of the first and estimated second parameters.

30 244. The receiver of claim 243 wherein the second signal comprises white noise.

245. The receiver of claim 243 wherein the first parameter comprises energy of the
signal and the second parameter comprises spectral characteristics of the signal.

35 246. The receiver of claim 245 wherein the signal generator comprises a power
controller to scale the second signal as a function of the energy, and a synthesis filter to filter the
second signal as a function of the estimated spectral characteristics.

1 247. The receiver of claim 246 wherein the signal generator comprises a noise source.
to generate the second signal.

5 248. The receiver of claim 247 wherein the noise source comprises a white noise
generator.

10 249. The receiver of claim 246 wherein the signal estimator comprises autocorrelation
logic to calculate autocorrelation coefficients for the signal, the filter logic generating linear
prediction coefficients as a function of the autocorrelation coefficients, the filtering of the second
signal by the synthesis filter being a function of the linear prediction coefficients.

15 250. A near end receiver for replicating far end background noise in a signal generated
by a far end transmitter, the near end receiver being adapted to receive the signal and a first
parameter of the signal, the near end receiver comprising:

a noise estimator to estimate a second parameter of the far end background noise
different from the first parameter of the far end background noise; and

a noise generator to modify a noise signal to replicate the far end background
noise as a function of the first and estimated second parameters.

20 251. The receiver of claim 250 wherein the noise generator comprises a noise source
to generate the noise signal.

25 252. The receiver of claim 250 wherein the noise source comprises a white noise
source.

25 253. The receiver of claim 250 wherein the first parameter comprises energy of the
signal and the second parameter comprises spectral characteristics of the signal.

30 254. The receiver of claim 253 wherein the noise generator comprises a power
controller to scale the noise signal as a function of the energy, and a synthesis filter to filter the
noise signal as a function of the estimated spectral characteristics.

30 255. The receiver of claim 254 wherein the noise estimator comprises autocorrelation
logic to calculate autocorrelation coefficients for the far end background noise, and filter logic
to generate linear prediction coefficients as a function of the autocorrelation coefficients, the
estimated spectral characteristics comprising the linear prediction coefficients.

35 256. A transmission system, comprising:
a far end transmitter which generates a far end signal having background noise,
the far end transmitter having a noise estimator to estimate a first parameter of the background
noise; and

1 a near end receiver coupled to the far end transmitter adapted to receive the signal
and the estimated first parameter, the near end receiver having a noise estimator to estimate a
second parameter different from the first parameter of the background noise of the received far
end signal, and a noise generator to modify a noise signal to replicate the background noise as
5 a function of the estimated first and second parameters.

257. The transmission system of claim 256 wherein the noise generator comprises a
noise source to generate the noise signal.

10 258. The transmission system of claim 257 wherein the noise source comprises a white
noise source.

259. The transmission system of claim 256 wherein the first parameter comprises
energy of the background noise and the second parameter comprises spectral characteristics of
the background noise.

15 260. The transmission system of claim 259 wherein the noise generator comprises a
power controller to scale the noise signal as a function of the estimated energy, and a synthesis
filter to filter the noise signal as a function of the estimated spectral characteristics.

20 261. The transmission system of claim 260 wherein the noise estimator comprises
autocorrelation logic to calculate autocorrelation coefficients for the background noise, and filter
logic to generate linear prediction coefficients as a function of the autocorrelation coefficients,
the estimated spectral characteristics comprising the linear prediction coefficients.

25 262. The transmission system of claim 256 further comprising a network coupling the
far end transmitter to the near end receiver.

30 263. The transmission system of claim 262 wherein the far end transmitter further
comprises a packetization engine to format the far end signal and the estimated first parameter
into a packet for transmission over the network, and the near end receiver further comprises a
depacketization engine to depacketize the far end signal and the estimated first parameter
received from the network before the estimated first parameter is coupled to the noise generator.

35 264. The transmission system of claim 262 further comprising a far end telephony
device coupled to the far end transmitter, and a near end telephony device coupled to the near end
receiver, the near end telephony device being adapted to receive the replicated far end
background noise generated by the noise generator.

265. The transmission system of claim 264 wherein the near end and far end telephony
devices each comprises a telephone.

1 266. The transmission system of claim 256 wherein the far end transmitter further comprises a voice activity detector to detect voice in the signal, the voice activity detector disabling the noise estimator when it detects voice.

5 267. A method for replicating far end background noise at a near end, comprising:
generating, at the far end, a far end signal having voice and the background noise;
estimating, at the far end, a first parameter of the background noise without voice;
transmitting the signal and the estimated first parameter of the background noise
from the far end to the near end;
estimating, at the near end, a second parameter different from the first parameter
10 of the background noise; and
modifying, at the near end, a noise signal to replicate the background noise as a
function of the estimated first and second parameters.

 268. The method of claim 267 wherein the noise signal comprises white noise.

15 269. The method of claim 267 wherein the first parameter comprises energy of the background noise and the second parameter comprises spectral characteristics of the background noise.

20 270. The method of claim 269 wherein the modification of the noise signal comprises scaling the noise signal as a function of the estimated energy, and filtering the noise signal as a function of the estimated spectral characteristics.

25 271. The method of claim 270 wherein the estimation of the spectral characteristics comprises estimating filter coefficients which model the spectral shape of the background noise, the filtering of the noise signal being a function of the estimated filter coefficients.

 272. The method of claim 271 wherein the estimation of the filter coefficients comprises calculating autocorrelation coefficients, the filter coefficients being a function of the autocorrelation coefficients.

30 273. The method of claim 271 wherein the estimation of the filter coefficients comprises calculating linear prediction coefficients, the filter coefficients being a function of the linear prediction coefficients.

35 274. The transmission system of claim 267 wherein the transmission of voice comprises transmitting voice over a network.

 275. The method of claim 267 further comprising transmitting the replicated far end background noise from the near end receiver to a telephony device.

1 276. The method of claim 275 wherein the near end telephony device comprises a telephone.

5 277. A method of compensating a received signal, comprising:
transmitting first and second signals, the first signal having a transmitted magnitude;
receiving the transmitted first and second signals, the received first signal having a received magnitude;
estimating the transmitted magnitude from the received first signal;
estimating a scaling factor as a function of the estimated transmitted magnitude
10 and the received magnitude; and
applying the estimated scaling factor to the received second signal.

15 278. The method of claim 277 wherein the transmitted first signal comprises a symbol representing a constellation point, and the transmitted magnitude estimation comprises quantizing the received symbol to its nearest constellation point.

 279. The method of claim 278 wherein the scaling factor estimation comprises dividing the transmitted magnitude estimation by the received magnitude.

20 280. The method of claim 277 further comprising modifying the estimated scaling factor by a second scale factor prior to applying the estimated scaling factor to the received second signal.

25 281. A method of compensating a signal, comprising:
quantizing a first signal;
estimating a scaling factor as a function of the quantized first signal and the first signal; and
applying the estimated scaling factor to a second signal.

30 282. The method of claim 281 wherein the first signal comprises a symbol representing a constellation point, and the first signal quantization comprises quantizing the symbol to its respective nearest constellation point.

 283. The method of claim 281 wherein the scaling factor estimation comprises dividing the quantized first signal by the first signal.

35 284. The method of claim 281 further comprising modifying the estimated scaling factor by a second scale factor prior to applying the estimated scaling factor to the second signal.

 285. A method of compensating a received signal, comprising:
estimating an expected magnitude of a first received signal;

1 estimating a scaling factor as a function of the expected magnitude estimation and
a magnitude of the received first signal; and
 applying the estimated scaling factor to a second received signal.

5 286. The method of claim 285 wherein the received first signal comprises a symbol
representing a constellation point, and the expected magnitude estimation comprises quantizing
the symbol to its nearest constellation point.

10 287. The method of claim 286 wherein the scaling factor estimation comprises dividing
the expected magnitude estimation by the received first signal magnitude.

15 288. The method of claim 285 further comprising modifying the estimated scaling
factor by a second scale factor prior to applying the estimated scaling factor to the received
second signal.

20 289. A signal compensator, comprising:
 a quantizer having an input and an output;
 a divider having a first input coupled to the quantizer input, a second coupled to
the quantizer output, and an output; and
 a multiplier having a first input adapted to receive a signal, a second input coupled
to the divider output, and an output coupled to the quantizer input.

25 290. The signal compensator of claim 289 wherein the quantizer comprises a slicer.

30 291. The signal compensator of claim 289 further comprising an averager between the
divider and the multiplier.

35 292. The signal compensator of claim 289 wherein the received signal comprises a
symbol representing a constellation point, and the quantizer is adapted to quantize the symbol
to its nearest constellation point.

40 293. The signal compensator of claim 289 further comprising a filter between the
divider and the multiplier.

45 294. The signal compensator of claim 293 wherein the filter comprises a non linear
filter.

50 295. The signal compensator of claim 294 wherein said non linear filter comprises a
hard limiter.

55 296. A data transmission system, comprising:
 a telephony device which outputs a signal; and

1 a signal processor coupled to the telephony device, the signal processor
comprising a quantizer having an input and an output, a divider having a first input coupled to
the quantizer input, a second coupled to the quantizer output, and an output, and a multiplier
5 having a first input adapted to receive a signal, a second input coupled to the divider output, and
an output coupled to the quantizer input.

297. The data transmission system of claim 296 wherein the quantizer comprises a
slicer.

10 298. The data transmission system of claim 296 wherein the signal processor further
comprises an averager between the divider and the multiplier.

299. The data transmission system of claim 296 wherein the received signal comprises
a symbol representing a constellation point, and the quantizer quantizes the symbol to its nearest
constellation point.

15 300. The data transmission system of claim 296 wherein the signal processor further
comprises a filter between the divider and the multiplier.

301. The data transmission system of claim 300 wherein the filter comprises a non
linear filter.

20 302. The data transmission system of claim 301 wherein the non linear filter comprises
a hard limiter.

303. The data transmission system of claim 301 wherein the telephony device
comprises a facsimile device.

304. The data transmission system of claim 301 wherein the telephony device
comprises a modem.

305. The data transmission system of claim 301 further comprising a public switched
30 telephone network coupled between the telephony device and the signal processor.

306. A method of compensating a signal, comprising:
receiving a plurality of first and second signals;
quantizing each of the received first signals;
estimating a scaling factor by dividing each of the quantized first signals with its
35 respective received first signal and averaging the resultant quotients; and
applying the estimated scaling factor to each of the second signals.

1 307. The method of claim 306 wherein the scale factor estimation further comprises
hard limiting each of the resultant quotients.

5 308. The method of claim 307 wherein the scale factor estimation further comprises
hard limiting each of the resultant quotients at a first value, the method further comprising
quantizing each of the scaled second signals, estimating a modified scaling factor by dividing
each of the quantized scaled second signals with its respective scaled second signal, hard limiting
each of the resultant quotients with a second value less than the first value and averaging the hard
limited resultant quotients, and applying the modified estimated scaling factor to the third signal.

10 309. A method of transmitting data, comprising:
receiving data from a network;
detecting network delay from the received data;
adding spoof data to the received data when the detected network delay exceeds
a threshold; and
15 transmitting the received data with the added spoof data to a telephony device.

310. The method of claim 309 wherein the telephony device comprises a fax device.

20 311. The method of claim 309 further comprising buffering the received data, and
wherein the network delay detection comprises monitoring an amount of the data buffered.

312. The method of claim 311 wherein the detected network jitter is inversely related
to the amount of the data buffered.

25 313. The method of claim 309 wherein the received data is formatted, and the added
spoof data comprises a format based on the format of the received data.

314. The method of claim 313 wherein the added spoof data format comprises HDLC
preamble flags when the received data format comprises handshake negotiation data.

30 315. The method of claim 313 wherein the added spoof data format comprises HDLC
preamble flags when the received data format comprises error correction mode fax image data.

316. The method of claim 315 further comprising assembling a frame having at least
a portion of the received data, and wherein the addition of spoof data further comprises inserting
the HDLC preamble flags at a frame boundary.

35 317. The method of claim 313 wherein the added spoof data format comprises zero fill
bits when the received data format comprises non-error correction mode fax image data.

1 318. The method of claim 317 wherein the non-error correction mode fax image data comprises image data followed by an end of line sequence, and the addition of spoof data comprises inserting the zero fill bits between the non-error correction mode fax image data signals and end of line sequence.

5 319. The method of claim 309 further comprising transmitting the data from a remote telephony device to the network before the data is received from the network.

10 320. The method of claim 319 wherein the telephony device and the remote telephony device each comprises a fax device.

15 321. A method of transmitting data, comprising:
receiving formatted data from a network; and
selectively spoofing a telephony device with spoof data, the spoof data having a format based on the format of the received data.

20 322. The method of claim 321 wherein the spoof data comprises format HDLC preamble flags when the received data format comprises handshake negotiation data.

25 323. The method of claim 321 wherein the spoof data format comprises HDLC preamble flags when the received data format comprises error correction mode fax image data.

30 324. The method of claim 323 further comprising assembling a frame having at least a portion of the received data, and wherein the spoofing of the telephony device comprises inserting the HDLC preamble flags at a frame boundary.

35 325. The method of claim 321 wherein the spoof data format comprises zero fill bits when the received data format comprises non-error correction mode fax image data.

40 326. The method of claim 325 wherein the non-error correction mode fax image data comprises image data followed by an end of line sequence, and the spoofing of the telephony device comprising inserting the zero fill bits between the non-error correction mode fax image data signals and end of line sequence.

45 327. The method of claim 321 wherein the telephony device comprises a fax device.

50 328. The method of claim 321 further comprising buffering the received data, and monitoring an amount of the buffered data, wherein the telephony device is spoofed only if the amount of the buffered data is below a threshold..

55 329. The method of claim 321 further comprising transmitting the data from a remote telephony device to the network before the formatted data is received from the network.

1 330. The method of claim 329 wherein the telephony device and the remote telephony
device each comprises a fax device.

5 331. A data exchange comprising spoofing logic to selectively generate spoof data for
a telephony device in response to formatted data received from a network, the spoof data being
generated with a format based on the format of the received data.

10 332. The data exchange of claim 331 wherein the spoofing logic is adapted to generate
spoof data with a format comprising HDLC preamble flags when the format of the received data
comprises handshake negotiation data.

15 333. The data exchange of claim 331 wherein the spoofing logic is adapted to generate
spoof data with a format comprising HDLC preamble flags when the format of the received data
comprises error correction mode fax image data.

20 334. The data exchange of claim 333 further comprising a data pump to assemble a
frame having at least a portion of the formatted data from the network, and to insert the HDLC
preamble flags at a frame boundary.

25 335. The data exchange of claim 331 wherein the spoofing logic is adapted to generate
spoof data with a format comprising zero fill bits when the format of the received data comprises
non-error correction mode fax image data.

30 336. The data exchange of claim 335 wherein the non-error correction mode fax image
data comprises image data followed by an end of line sequence, the data exchange further
comprising a data pump to insert the zero fill bits between the non-error correction mode fax
image data signals and end of line sequence.

35 337. The data exchange of claim 331 further comprising a buffer to buffer the
formatted data received from the network, the spoofing logic being adapted to generate the
spoofing data when an amount of the formatted data in the buffer level is below a threshold.

40 338. A data exchange comprising spoofing logic to detect network delay from data
received from a network and to generate spoof data when the detected network delay exceeds a
threshold.

45 339. The data exchange of claim 338 further comprising a buffer to buffer the data
received from the network, the spoofing logic being adapted to detect network delay based on an
amount of the data in the buffer.

 340. The data exchange of claim 339 wherein the detected network delay is inversely
related to the amount of the data in the buffer.

1 341. The data exchange of claim 338 wherein the received data is formatted, and the spoofing logic is adapted to format the spoof data based on the format of the received data.

5 342. The data exchange of claim 341 wherein the spoofing logic is adapted to generate spoof data with a format comprising HDLC preamble flags in response to the received data format being handshake negotiation data.

10 343. The data exchange of claim 341 wherein the spoofing logic is adapted to generate spoof data with a format comprising HDLC preamble flags in response to the received data format being error correction mode fax image data.

15 344. The data exchange of claim 343 further comprising a data pump to assemble a frame having at least a portion of the data received from the network, and to insert the HDLC preamble flags at a frame boundary.

20 345. The data exchange of claim 341 wherein the spoofing logic is adapted to generate spoof data with a format comprising zero fill bits when the received data format comprises non-error correction mode fax image data, and

25 346. The data exchange of claim 345 wherein the non-error correction mode fax image data comprises image data followed by an end of line sequence, the data exchange further comprising a data pump to insert the zero fill bits between the non-error correction mode fax image data signals and end of line sequence.

30 347. A signal transmission system, comprising:
a network;
a first telephony device;
a second telephony device in a communication with the first telephony device through the network;

35 a first data exchange coupling the first telephony device to the network, the first data exchange comprising first spoofing logic to detect network delay in a first path from first data received from the second telephony device, the first spoofing logic being adapted to generate first spoof data for the first telephony device when the detected network delay in the first path exceeds a first threshold; and

 a second data exchange coupling the second telephony device to the network, the second data exchange comprising second spoofing logic to detect network delay in a second path from second data received from the first telephony device, the second spoofing logic being adapted to generate second spoof data for the second telephony device when the detected network delay in the second path exceeds a second threshold.

 348. The signal transmission system of claim 347 wherein the first data exchange comprises a first buffer to buffer the first data, the detected network delay in the first path being

1 based on an amount of the first data in the first buffer, and the second data exchange comprises
a second buffer to buffer the second data, the detected network delay in the second path being
based on an amount of the second data in the second buffer.

5 349. The signal transmission system of claim 348 wherein the detected network delay
in the first path is inversely related to the amount of the first data in the first buffer, and the
detected network delay in the second path is inversely related to the amount of the second data
in the second buffer.

10 350. The signal transmission system of claim 347 wherein the first data and the second
data are formatted, the first and second spoofing logic being adapted to generate its respective
spoofing data with a format based on the format of the first data and the second data.

15 351. The signal transmission system of claim 350 wherein the first and second spoofing
logic are each adapted to generate its respective spoofing data with a format comprising HDLC
preamble flags when the first and second data format comprises handshake negotiation data.

352. The signal transmission system of claim 350 wherein the first spoofing logic is
adapted to generate first spoofing data with a format comprising HDLC preamble flags when the
first data format comprises error correction mode fax image data.

20 353. The signal transmission system of claim 352 wherein the first data exchange
further comprises a data pump to assemble a frame having at least a portion of the first data
received from the second telephony device, and to insert the HDLC preamble flags at a frame
boundary.

25 354. The signal transmission system of claim 350 wherein the first spoofing logic is
adapted to generate the first spoof data with a format comprising zero fill bits when the first data
format comprises non-error correction mode fax image data.

30 355. The signal transmission system of claim 354 wherein the non-error correction
mode fax image data comprises image data followed by an end of line sequence, the first data
exchange further comprising a data pump to insert the zero fill bits between the non-error
correction mode fax image data signals and end of line sequence.

356. The signal transmission system of claim 347 wherein the first and second
telephony devices each comprises a fax device.

35 357. A signal transmission system, comprising:
a network;
a first telephony device;

1 a second telephony device in communication with the first telephony device through the network;

5 a first data exchange coupling the first telephony device to the network, the first data exchange comprising first spoofing logic to selectively generate first spoof data for the first telephony device in response to formatted first data received from the second telephony device, the first spoof data being a function of the first data format; and

10 a second data exchange coupling the second telephony device to the network, the second data exchange comprising second spoofing logic to selectively generate second spoof data for the second telephony device in response to formatted second data received from the first telephony device, the second spoof data being a function of the second data format.

15 358. The signal transmission system of claim 357 wherein the first spoofing logic is adapted to generate the first spoofing data with a format comprising HDLC preamble flags in response to the formatted first data being handshake negotiation data.

20 359. The signal transmission system of claim 357 wherein the the first spoofing logic is adapted to generate the first spoof data with a format comprising HDLC preamble flags in response to the formatted first data being error correction mode fax image data.

25 360. The signal transmission system of claim 359 wherein the first data exchange further comprising a data pump to assemble a frame having at least a portion of the formatted first data received from the second telephony device, and to insert the HDLC preamble flags at a frame boundary.

30 361. The signal transmission system of claim 357 the first spoofing logic is adapted to generate the first spoof data with a format comprising zero fill bits in response to the formatted first data being non-error correction mode fax image data.

35 362. The signal transmission system of claim 361 wherein the non-error correction mode fax image data comprises image data followed by an end of line sequence, the first data exchange further comprising a data pump to insert the zero fill bits between the non-error correction mode fax image data signals and end of line sequence.

40 363. The signal transmission system of claim 357 wherein the first data exchange further comprises a first buffer to buffer the formatted first data received from the second telephony device, the first spoofing logic being adapted to generate the first spoof data when an amount of the received formatted first data in the buffer level is below a first threshold, and the second data exchange further comprises a second buffer to buffer the formatted second data received from the first telephony device, the second spoofing logic being adapted to generate the second spoof data when an amount of the received formatted first data in the buffer level is below a second threshold

1 364. The signal transmission system of claim 357 wherein the first and second telephony devices each comprises a fax device.

5 365. A method for detecting a tone in a composite signal having a plurality of components comprising:

 separating one of the components from the signal; and
 detecting from a portion of the separated component whether the separated component comprises the tone.

10 366. The method of claim 365 further comprising formatting the separated component into first and second frames, the first preceding the second frame in time, each of the first and second frames having first and second portions, and wherein the tone detection comprises detecting from the second portion of the first frame whether the separated component comprises a tone.

15 367. The method of claim 366 wherein the first portion of the frame precedes the second portion of the frame in time.

20 368. The method of claim 365 further comprising formatting the samples into first and second frames, the first frame preceding the second frame in time, each of the first and second frames having first and second portions, and wherein the tone detection comprises detecting from the second portion of the first frame whether the signal comprises a tone.

 369. The method of claim 368 wherein the first portion precedes the second portion in time for each of the first and second frames.

25 370. The method of claim 365 further comprising formatting the separated component into first and second frames, the first frame preceding the second frame in time, each of the first and second frames having first and second portions, the first portion of the frame precedes the second portion of the frame in time for each of the first and second frames, and bypassing the tone detection for the first portion of the second frame if the tone detection does not detect the tone in the second portion of the first frame.

30 371. A method of dual tone signal detection in a composite signal having first and second components, comprising:

 separating the composite signal into its first and second components;
 detecting from a portion of the first component whether the first component comprises a first one of the dual tones; and
35 detecting from a portion of the second component whether the second component comprises a second one of the dual tones.

1 372. The method of claim 371 further comprising formatting the first component into a frame having first and second portions, and wherein the detection of the first one of the dual tones comprises detecting from the second portion of the frame whether the first component comprises the first one of the dual tones.

5 373. The method of claim 372 wherein the first portion of the frame precedes the second portion of the frame in time.

10 374. The method of claim 371 further comprising formatting the first component into first and second frames, the first frame preceding the second frame in time, each of the first and second frames having first and second portions, and wherein the detection of the first one of the dual tones comprises detecting from the second portion of the first and second frames whether the first component comprises.

15 375. The method of claim 374 wherein the first portion precedes the second portion in time for each of the first and second frames.

20 376. The method of claim 371 further comprising formatting the first component into first and second frames, the first frame preceding the second frame in time, each of the first and second frames having first and second portions, the first portion of the frame precedes the second portion of the frame in time for each of the first and second frames, and bypassing the detection of the first one of the dual tones for the first portion of the second frame if the detection for the first one of the dual tones does not detect the first one of the dual tones in the second portion of the first frame.

25 377. A method of detecting a tone in a composite signal having first and second components, comprising:

 separating the composite signal into its first and second components;
 determining frequency for each of the first and second components; and
 detecting as a function of the determined frequency for each of the first and second components whether either of the first and second components comprises the tone.

30 378. The method of claim 377 further comprising estimated a characteristic different from the frequency for each of the first and second components, wherein the true detection is further a function of the estimated characteristic.

 379. The method of claim 378 wherein the characteristic comprises power.

35 380. The method of claim 379 wherein the tone detection further comprises comparing the estimated power for each of the first and second components to a threshold.

1 381. The method of claim 377 wherein the tone detection comprises comparing the determined frequency of each of the separated first and second components to a plurality of frequency ranges to determine whether either of the first and second components comprises the tone.

5 382. The method of claim 377 wherein the composite signal separation comprises bandpass filtering the composite signal into its first and second components.

10 383. The method of claim 382 wherein the frequency determination comprises down-sampling the filtered first and second components.

15 384. The method of claim 382 wherein the composite signal bandpass filtering comprises complex filtering.

20 385. The method of claim 377 wherein the frequency determination comprises converting the separated first and second components into complex signals.

25 386. The method of claim 379 wherein the tone detection further comprises comparing a ratio of the power estimation for the first and second components to a threshold.

30 387. A system for detecting a tone in a composite signal having a plurality of components comprising:

 a filter to separate one of the components from the composite signal; and
 a detector to detect from a portion of the separated component whether the separated component comprises the tone.

35 388. The tone detection system of claim 387 wherein the separated component comprise first and second portions, the tone detection system further comprising a state machine to invoke the detector to detect the tone in the second portion of the separated component.

40 389. The tone detection system of claim 388 wherein the first portion precedes the second portion in time.

45 390. The tone detection system of claim 387 wherein the separated component comprise first and second frames, the first frame preceding the second frame in time, each of the first and second frames having first and second portions, the tone detection system further comprising a state machine to invoke the detector to detect the tone in the second portion of the first frame, and to invoke the detector to process the tone in the first portion of the second frame only if the detector detects the tone in the second portion of the first frame.

50 391. The tone detection system of claim 390 wherein the first portion precedes the second portion in time for each of the first and second frames.

1 392. A system of detecting a dual tone in a composite signal having first and second components, comprising:

 a first bandpass filter to separate the first component from the composite signal;
 a second bandpass filter to separate the second component from the composite

5 signal;

 a first detector to determine frequency of the separated first component;
 a second detector to determine frequency of the separated second component;
 a first comparator to compare the frequency of the first component to at least one
of a plurality of frequency ranges to determine whether the separated first component comprises
one of the dual tones; and

10 a second comparator to compare the frequency of the separated second component
to at least one of the frequency ranges to determine whether the second component comprises the
other one of the dual tones.

15 393. The tone detection system of claim 392 further comprising a first power estimator
to the estimate power of the separated first component, the first comparator further comparing
the estimated power of the separated first component to a power threshold, the determination of
whether the separated first signal comprises said one of the tones being further a function of the
comparison.

20 394. The tone detection system of claim 392 further comprising first and second power
estimators each estimating power of a respective one of the first and second separated
components, and a twist estimator to compare a ratio of the estimated power for the first and
second components, the determination of whether the composite signal comprises the dual tone
being further a function of the comparison.

25 395. The dual detection system of claim 392 wherein each of the first and second
comparators comprises a frequency calculator that estimates a mean frequency deviation from
one of a plurality of frequencies for each of the separated first and second components and
compares the mean of each of the separated first and second components to a respective
threshold.

30 396. The dual detection system of claim 392 further comprising a first summer to
convert the separated first component to a first component to a first complex signal prior to the
frequency determination by the first detector, and a second summer to convert the separated
second component to a second complex signal prior to the frequency determination by the second
detector.

35 397. The dual tone detection system of claim 392 wherein each of the first and second
bandpass filters comprises complex filters.

1 398. The dual tone detection system of claim 392 further comprising a first
downsampler to downsample the separated first component prior to the frequency determination
by the first detector, and a second downsampler to downsample the separated second component
prior to the frequency determination by the second detector.

5 399. A data transmission system, comprising:
a telephony device having a composite signal output comprising a plurality of
components; and
a signal processing system coupled to the telephony device, the signal processing
system comprising a detector to separate one of the components from the composite signal and
10 detect from a portion of the separated component whether the separated component comprises
a tone.

15 400. The data transmission system of claim 399 wherein the separated component
comprises first and second portions, and the signal processing system further comprising a state
machine to invoke the detector to detect the tone in the second portion of the samples.

 401. The data transmission system of claim 400 wherein the first portion precedes the
second portion in time.

20 402. The data transmission system of claim 399 wherein the separated component
comprise first and second frames, the first frame preceding the second frame in time, each of the
first and second frames having first and second portions, and wherein the signal processing
system further comprising a state machine to invoke the detector to detect the tone in the second
portion of the first frame, and to invoke the detector to process the tone in the first portion of the
second frame only if the detector detects a tone in the second portion of the first frame.

25 403. The data transmission system of claim 402 wherein the first portion precedes the
second portion in time for each of the first and second frames.

30 404. The data transmission system of claim 399 wherein the telephony device
comprises a telephone.

35 405. A system for transmitting a dual tone, comprising:
a telephony device having a composite signal output comprising first and second
components; and
a signal processing system coupled to the telephony device, the signal processing
system comprising,
a first bandpass filter to separate the first component from the composite
signal,
a second bandpass filter to separate the second component from the
composite signal,

1 a first power estimator to estimate power of the separated first component,
 a second power estimator to estimate power of the separated second
 component,

 a first detector to determine frequency of the separated first component,
5 a second detector to determine frequency of the separated second
 component, a first comparator to compare the estimated power and determined
 frequency of the first component respectively to a power threshold and frequency
 range to determine whether the first component comprises one of the dual tones,
 and

 a second comparator to compare the estimated power and determined
10 frequency of the second component respectively to a power threshold and
 frequency range to determine whether the second component comprises the other
 one of the dual tones.

406. The data transmission system of claim 405 wherein each of the first and second
15 comparators comprises a frequency calculator that estimates a mean deviation to one of a
 plurality of frequencies for each of the separated first and second components and compares the
 estimated mean for each of the separated first and second components to a respective threshold.

407. The data transmission system of claim 405 wherein the signal processing system
20 further comprising a first summer to convert the separated first component to a first complex
 signal prior to the first power estimation and frequency determination, and a second summer to
 convert the separated second component to a second complex signal prior to the second power
 estimation and frequency determination.

408. The data transmission system of claim 405 wherein each of the first and second
25 bandpass filters comprises complex filters.

409. The data transmission system of claim 405 wherein the signal processing system
 further comprising a first downsampler to downsample the separated first component prior to the
 first power estimation and frequency determination, and a second downsampler to downsample
30 the separated second component prior to the second power estimation and frequency
 determination

410. A method of conditioning a composite signal, the composite signal being formed
 by introducing at least a portion of a first signal into a second signal, comprising:
 estimating a characteristic of at least one of said first and composite signals; and
 selectively conditioning the composite signal, the selection of whether to
35 condition the composite signal being based on the estimated characteristic.

411. The method of claim 410 wherein the characteristic estimation comprises
 estimating a power level of the first signal, and estimating an echo return loss between the first

1 signal and the composite signal, and wherein the composite signal is conditioned echo if the
estimated power level of the first signal minus the echo return loss is greater than a threshold.

5 412. The method of claim 410 wherein the conditioning of the composite signal
comprises adaptively filtering the first signal, and recovering the second signal by subtracting the
filtered first signal from the composite signal.

413. The method of claim 412 further comprising selectively limiting filter adaptation,
the selection of whether to limit the filter adaptation being based on the estimated characteristic.

10 414. The method of claim 413 wherein the filter adaptation is limited by disabling the
filter adaptation.

15 415. The method of claim 412 wherein the characteristic estimation comprises
estimating a return loss between the composite signal and the first signal, estimating a return loss
enhancement, the return loss enhancement comprising a reduction in power of the composite
signal due to the signal conditioning in the absence of the second signal, and wherein the
conditioning of the composite signal further comprises adjusting the filter adaptation as a
function of at least one of the estimated return loss and the estimated return loss enhancement.

20 416. The method of claim 412 wherein the characteristic estimation, comprises:
estimating a first power level of the first signal;
estimating a second power level of the composite signal;
estimating a return loss between the composite signal and the first signal by
dividing the first power level by the second power level;
estimating a third power level of the recovered second signal; and
25 estimating a return loss enhancement by dividing the second power level by the
third power level;
wherein the conditioning of the composite signal further comprises adjusting the
filter adaptation as a function of at least one of the return loss and return loss enhancement.

30 417. The method of claim 412 further comprising processing the recovered second
signal when information is detected in the first signal but not in the second signal.

418. The method of claim 417 wherein the recovered second signal is processed by
attenuation.

35 419. The method of claim 417 wherein the processing of the recovered second signal
is non-linear.

1 420. A method of cancelling a far end echo from a near end signal, comprising:
 estimating a characteristic of at least one of a far end signal and the near end
 signal; and

5 selectively cancelling the echo from the near end signal, the selection of whether
 to cancel the echo from the near end signal being based on the estimated characteristic.

10 421. The method of claim 420 wherein the characteristic estimation comprises
 estimating a power level of the far end signal, and estimating an echo return loss between the far
 end signal and the near end signal, and wherein the echo is cancelled from the near end signal if
 the estimated power level of the far end signal minus the echo return loss is greater than a
 threshold.

15 422. The method of claim 420 wherein the characteristic estimation comprises
 estimating a power level of the far end signal, estimating an echo return loss between the far end
 signal and the near end signal, and estimating a power level of the near end signal, wherein the
 selection of whether to cancel the echo from the near end signal is based on the estimated power
 levels and the estimated echo return loss.

20 423. The method of claim 420 wherein the echo cancellation comprises adaptively
 filtering the far end signal and subtracting the filtered far end signal from the near end signal.

25 424. The method of claim 423 further comprising selectively limiting filter adaptation,
 the selection of whether to limit the filter adaptation being based on the estimated characteristic.

30 425. The method of claim 424 wherein the filter adaptation is limited by disabling the
 filter adaptation.

35 426. The method of claim 420 wherein the characteristic estimation comprises
 estimating a power level of the far end signal, estimating an echo return loss between the far end
 signal and the near end signal, and estimating a power level for noise on the near end signal
 without the echo, and wherein the echo is canceled from the near end signal when the power level
 of the far end signal minus the echo return loss is greater than both a threshold of hearing and
 the power level for the noise minus about 10 dB.

40 427. The method of claim 423 wherein the characteristic estimation comprises
 estimating an echo return loss between the far end signal and the near end signal, and estimating
 an echo return loss enhancement between the near end signal and the near end signal without the
 echo, and wherein filter adaptation is a function of at least one of the echo return loss and echo
 return loss enhancement.

1 428. The method of claim 427 wherein the filter adaptation comprises using an adaptation step size of one-fourth when the echo return loss enhancement is in the range of 0-9 dBm.

5 429. The method of claim 427 wherein the filter adaptation comprises using an adaptation step size of 1/32 when a combination of the estimated echo return loss and the echo return loss enhancement is greater than 33-36 dB.

10 430. The method of claim 427 wherein the filter adaptation comprises using an adaptation step size of 1/16 when a combination of the estimated echo return loss and the echo return loss enhancement is in the range of 23-33 dB.

15 431. The method of claim 423 further comprising detecting information in the near end signal, wherein the filter adaptation comprises limiting the filter adaptation when the information is detected and the filter adaptation is converged.

20 432. The method of claim 431 wherein the limiting of the filter adaption comprises disabling the filter adaption.

25 433. The method of claim 423 wherein the filter adaptation is limited when the filter adaptation has been active for a period longer than one second from an off hook transition of a telephony device connected between the far end signal and the near end signal.

30 434. The method of claim 423 wherein the filter adaptation is limited when the filter adaptation has been active for a period longer than one second after filter adaptation initialization.

35 435. The method of claim 431 wherein the filter adaptation comprises using an adaptation step size of 1/32 when the information is detected and the filter adaptation is not converged.

40 436. The method of claim 423 wherein the characteristic estimation further comprises estimating a power level of the far end signal, and estimating a power level for noise on the near end signal without the echo, and wherein the filter adaptation comprises using an adaptation step size of 1/4 when the estimated power level of the far end signal exceeds the estimated power level of the noise by at least 24 dB.

45 437. The method of claim 423 wherein the characteristic estimation comprises estimating a power level of the far end signal, and estimating a power level for noise on the near end signal without the echo, and wherein the filter adaptation comprises using an adaptation step size of 1/8 when the estimated power level of the far end signal exceeds the estimated power level of the noise by at least 18 dB.

1 438. The method of claim 423 wherein the characteristic estimation further comprises
estimating a power level of the far end signal, and estimating a power level for noise on the near
end signal without the echo, and wherein the filter adaptation comprises using an adaptation step
size of 1/16 when the estimated power level of the far end signal exceeds the estimated power
5 level of the noise by at least 9 dB.

 439. The method of claim 420 further comprising detecting information in the far end
signal, detecting information in the near end signal, and processing the near end signal when
information is detected in the far end signal and not in the near end signal.

10 440. The method of claim 439 wherein the near end is processed by attenuation.

 441. The method of claim 439 wherein the processing of the near end signal is non-
linear.

15 442. The method of claim 420 wherein the characteristic estimation comprises
estimating a power level of the far end signal, estimating a power level of the near end signal,
estimating a power level of a near end signal without the echo, estimating a power level of noise
on the far end signal, and selectively non linear processing the near end signal, the selection as
to whether to non linear process the near end signal being based on the estimated power levels.

20 443. The method of claim 442 further comprising setting a first decision variable as a
function of the estimated power level of the far end signal, setting a second decision variable as
a function of the power level of the near end signal without the echo, setting a third decision
variable as a function of the estimated power level of the far end signal and the near end signal
without the echo, wherein the is near end signal is non linear processed when at least of the two
25 decision variables meet a respective criteria.

 444. The method of claim 443 wherein the first decision variable is set when the
estimated power level of the far end signal is at least 6 dB greater than the estimated power level
of the noise on the far end signal, and the estimated power level of the far end signal minus an
estimated echo return loss between the far end signal and the near end signal is at least 6 dB
30 greater larger than the estimated power level of the near end signal.

 445. The method of claim 442 wherein the second decision variable is set when the
estimated power level of the near end signal without the echo is at least 9 dB less than the
estimated power level of the near end signal.

35 446. The method of claim 442 wherein the third decision variable is set when the
estimated power level of the far end signal minus the estimated power level of the near end signal
without the echo is greater than a threshold power level.

1 447. A signal conditioner for conditioning a composite signal, the composite signal
being formed by introducing at least a portion of a first signal into a second signal, comprising:
a canceller to recover the second signal from the composite signal; and
a bypass to selectively enable the canceller.

5 448. The signal conditioner of claim 447 further comprising a power estimator to
estimate a maximum power level and an average power level of the first signal, and adaptation
logic to estimate a return loss between the first signal and the composite signal, wherein the
bypass enables the canceller as a function of at least one of the estimated maximum power level,
10 the estimated average power level, and the estimated return loss.

15 449. The signal conditioner of claim 448 wherein the bypass enables the canceller when
the estimated maximum power level of the first signal minus the estimated return loss is greater
than a threshold.

20 450. The signal conditioner of claim 448 further comprising a second power estimator
to estimate an average power level of the composite signal, wherein the adaptation logic
estimates the return loss by dividing the estimated average power level of the first signal by the
estimated average power level of the composite signal.

25 451. The signal conditioner of claim 450 wherein the bypass enables the canceller when
the estimated maximum power level of the first signal minus the estimated return loss is at least
8 dB greater than the estimated power level of the composite signal.

30 452. The signal conditioner of claim 447 wherein the canceller further comprises an
adaptive filter to filter the first signal, and a combined operator to subtract the filtered first signal
from the composite signal to recover the second signal.

35 453. The signal conditioner of claim 452 further comprising a processor, and adaptation
logic which invokes the processor to suppress the recovered second signal when information is
detected in the first signal but not in the composite signal.

40 454. The signal conditioner of claim 453 wherein the processor comprises a non-linear
processor.

45 455. The signal conditioner of claim 454 wherein the filter adaptation is limited by
disabling the adaptation of the adaptive filter.

50 456. The signal conditioner of claim 453 wherein the information includes voice.

55 457. The signal conditioner of claim 452 further comprising a first power estimator to
estimate a maximum power level of the first signal, a second power estimator to estimate a noise

1 power level for the recovered second signal, and adaptation logic to estimate a return loss
between the first signal and the composite signal, wherein the bypass enables the canceller when
the estimated maximum power level of the first signal minus the estimated return loss is greater
than both a threshold of hearing and the estimated power level of the noise of the recovered
5 second signal minus 8 dB.

458. The signal conditioner of claim 452 further comprising a filter adapter to adjust
the adaptation of the adaptive filter.

10 459. The signal conditioner of claim 458 wherein the filter adapter limits the adaptation
of the adaptive filter when the bypass does not enable the canceller.

15 460. The signal conditioner of claim 458 further comprising adaptation logic to
estimate a return loss between the first signal and the composite signal, and a return loss
enhancement between the composite signal and the recovered second signal, the filter adapter
adjusting the adaptation of the adaptive filter as a function of the estimated return loss and the
estimated return loss enhancement.

20 461. The signal conditioner of claim 460 further comprising a first power estimator to
estimate a maximum power level and an average power level of the first signal, a second power
estimator to estimate an average power level of the composite signal, a third power estimator to
estimate an average power level and a noise power level for the recovered second signal, wherein
the adaptation logic estimates the return loss and the return loss enhancement as a function of the
estimated power levels.

25 462. The signal conditioner of claim 461 wherein the adaptation logic estimates the
return loss by dividing the average power level of the first signal by the average power level of
the composite signal.

30 463. The signal conditioner of claim 461 wherein the adaptation logic estimates the
return loss enhancement by dividing the average power of the composite signal by the average
power of the recovered second signal.

35 464. The signal conditioner of claim 461 wherein the filter adapter causes the adaptive
filter to have a filter adaptation step size of 1/4 when the estimated average power level of the
first signal is 24 dB greater than the estimated power level of the noise of the recovered second
signal.

465. The signal conditioner of claim 461 wherein the filter adapter causes the adaptive
filter to have a filter adaptation step size of about 1/8 when the estimated average power level of
the first signal is 18 dB greater than the estimated power level of the noise on the recovered
second signal.

1 466. The signal conditioner of claim 461 wherein the filter adapter causes the adaptive filter to have a filter adaptation step size of 1/16 when the estimated average power level of the first signal is 9 dB greater than the estimated power level of the noise on the recovered second signal.

5 467. The signal conditioner of claim 460 wherein the filter adapter causes the adaptive filter to have an adaptation step size of 1/16 when a combination of the estimated return loss and the estimated return loss enhancement is in the range of about 23-33 dB.

10 468. The signal conditioner of claim 460 wherein the adaptation logic limits the filter adapter when the adaptation logic detects information in the composite signal and the adaptive filter is converged.

 469. The signal conditioner of claim 468 wherein the information includes voice.

15 470. The signal conditioner of claim 460 wherein the adaptation logic limits the adaptation of the adaptive filter when the adaptive filter has been active for a period longer than one second after an off hook transition of a telephony device coupled between the first signal and the composite signal.

20 471. The signal conditioner of claim 460 wherein the adaptation logic limits the adaptation of the adaptive filter when the adaptive filter has been active for a period longer than one second after the adaptive filter is initialized.

25 472. The signal conditioner of claim 460 wherein the filter adapter causes the adaptive filter to have an adaptation step size of 1/32 when the adaptation logic detects information in the composite signal and the adaptive filter is not converged.

 473. The signal conditioner of claim 460 wherein the filter adapter causes the adaptive filter to have an adaptation step size of one-fourth when the estimated return loss enhancement is in the range of 0-9 dBm.

30 474. The signal conditioner of claim 460 wherein the filter adapter causes the adaptive filter to have an adaptation step of 1/32 when a combination of the estimated return loss and the estimated return loss enhancement is greater than 33 dB.

35 475. A method of managing resources of a system, comprising:
 processing data;
 estimating data processing complexity; and
 reducing the data processing complexity when the estimated complexity exceeds
a threshold.

1 476. The method of claim 475 wherein the data processing comprises canceling echos from the data, and the data processing complexity reduction comprises bypassing the echo cancellation.

5 477. The method of claim 475 wherein the data processing comprises adaptively canceling the echos from the data, and the data processing complexity reduction further comprises reducing the complexity of the echo cancellation adaption.

10 478. The method of claim 477 wherein the reduction of the echo cancellation adaption complexity comprises disabling the echo cancellation adaption.

15 479. The method of claim 475 wherein the data processing comprises encoding the data, and the data processing complexity reduction comprises reducing the complexity of the data encoding.

20 480. The method of claim 479 wherein the data encoding comprises searching an adaptive codebook, and the data encoding complexity reduction comprises reducing complexity of the adaptive codebook search.

25 481. The method of claim 479 wherein the data encoding comprises searching an adaptive codebook, and the data encoding complexity reduction comprises bypassing the adaptive codebook search.

30 482. The method of claim 479 wherein the data encoding comprises performing an excitation search, and the data encoding complexity reduction comprises reducing the complexity of the excitation search.

35 483. The method of claim 482 wherein the excitation search comprises a fixed excitation search.

40 484. The method of claim 475 wherein the data processing comprises decoding the data, and the data processing complexity reduction comprises reducing the complexity of the data decoding.

45 485. The method of claim 484 wherein the data processing further comprises filtering the decoded data, and the data decoding complexity reduction comprises disabling the data filtering.

50 486. The method of claim 475 wherein the data processing complexity reduction comprises reducing the data processing complexity to one of a plurality complexity reduction levels based on a magnitude in which the estimated data processing complexity exceeds the threshold.

1 487. The method of claim 475 wherein the data comprises a near and far end signal,
and the data processing comprises canceling echos on a near end signal, the echos being
introduced into the near end signal by a far end signal, and the data processing complexity
estimation comprising estimating the data processing complexity based on power of the far end
5 signal and power of the echo canceled near end signal.

 488. The method of claim 475 wherein the data comprises voice including active voice
and silent periods, and the data processing comprises encoding the data, the data encoding
including detecting active voice, the data processing complexity estimation comprising
estimating the data processing complexity based on the active voice detection,
10

 489. The method of claim 488 wherein the data processing complexity reduction
comprises reducing the complexity of the data encoding.

 490. The method of claim 475 wherein the data comprises first and second frames, the
first frame preceding the second frame in time, and wherein the data processing complexity
estimation for the second frame is based on the data in the first frame.
15

 491. A method of managing resources of a system, comprising:
 performing a plurality of system functions on data;
 estimating average complexity of each the system functions;
20 summing the estimated average complexity of each of the system functions; and
 reducing complexity of at least one of the system functions when the sum of the
estimated average complexities exceeds a threshold.

 492. The method of claim 491 wherein said at least one of the system functions
comprises canceling echos from the data, and the complexity reduction comprises bypassing the
echo cancellation.
25

 493. The method of claim 491 wherein said at least one of the system functions
comprises adaptively canceling echos from the data, and the complexity reduction comprises
reducing the complexity of the echo cancellation adaption.
30

 494. The method of claim 493 wherein the reduction of the echo cancellation adaption
complexity comprises disabling the echo cancellation adaption.

 495. The method of claim 491 wherein said at least one of the system functions
comprises encoding the data, and the complexity reduction comprises reducing the complexity
of the data encoding.
35

1 496. The method of claim 495 wherein the data encoding comprises searching an
adaptive codebook, and the data encoding complexity reduction comprises reducing complexity
of the adaptive codebook search.

5 497. The method of claim 495 wherein the data encoding comprises searching an
adaptive codebook, and the data encoding complexity reduction comprises bypassing the adaptive
codebook search.

10 498. The method of claim 495 wherein the data encoding comprises performing an
excitation search, and the data encoding complexity reduction comprises reducing the complexity
of the excitation search.

 499. The method of claim 498 wherein the excitation search comprises a fixed
excitation search.

15 500. The method of claim 491 wherein the complexity reduction comprises reducing
the complexity of said at least one of the system functions such that system complexity is reduced
to one of a plurality complexity reduction levels based on a magnitude in which the sum of the
estimated average complexities exceeds the threshold.

20 501. The method of claim 491 further comprising detecting when system complexity
exceeds the threshold after the complexity reduction of said at least one of the system functions,
and further reducing the complexity of said at least one of the system functions or reducing the
complexity of at least a second one of the system functions when the system complexity exceeds
the threshold.

25 502. The method of claim 491 wherein the data comprises voice including active voice
and silent periods, and said at least one of the system functions comprises encoding the data, the
data encoding including detecting active voice, the average complexity estimation of the data
encoding being based on the active voice detection.

30 503. The method of claim 502 wherein the data processing complexity reduction
comprises reducing the complexity of the data encoding.

 504. The method of claim 491 wherein said at least one of the system functions
comprises decoding the data, and the complexity reduction comprises reducing the complexity
of the data decoding.

35 505. The method of claim 504 wherein the data decoding comprises filtering the
decoded data, and the data decoding complexity reduction comprises disabling data filtering.

1 506. The method of claim 491 wherein the data comprises first and second frames, the first frame preceding the second frame in time, and wherein the data processing complexity estimation for each of the system functions for the second frame is based on the data in the first frame.

5 507. A data transmission system, comprising:
a telephony device which outputs a signal; and
a signal processor coupled to the telephony device, the signal processor comprising a resource manager that estimates signal processor complexity and reduces the signal processor complexity when the estimated complexity exceeds a threshold.

10 508. The data transmission system of claim 507 wherein the resource manager reduces the signal processor complexity to one of a plurality of complexity reductions levels based on a magnitude in which the estimated signal processor complexity exceeds the threshold.

15 509. The data transmission system of claim 507 wherein the signal processor comprises an echo canceller, and the resource manager reduces the signal processor complexity by bypassing the echo canceller.

20 510. The data transmission system of claim 507 wherein the signal processor comprises an adaptive echo canceller, and the resource manager reduces the signal processor complexity by reducing the adaption complexity of the echo canceller.

 511. The data transmission system of claim 510 wherein the reduction of the echo canceller adaption complexity comprises disabling the adaption of the echo canceller.

25 512. The data transmission system of claim 507 wherein the signal processor comprises an encoder, and the resource manager reduces the signal processor complexity with encoder complexity reductions.

30 513. The data transmission system of claim 512 wherein the encoder searches an adaptive codebook, and the resource manager reduces the encoder complexity by reducing search complexity of the adaptive codebook.

 514. The data transmission system of claim 512 wherein the encoder includes an adaptive codebook, and the resource manager reduces the encoder complexity with an adaptive codebook bypass.

35 515. The data transmission system of claim 512 wherein the encoder performs an excitation search, and the resource manager reduces the encoder complexity by reducing complexity of the excitation search.

1 516. The method of claim 515 wherein the excitation search comprises a fixed
excitation search.

5 517. The data transmission system of claim 507 wherein the signal processor comprises
an echo canceller to cancel echos from a near end signal, the echos being introduced into the near
end signal by a far end signal, and the resource manager estimates the signal processor
complexity based on power of the far end signal and power of the echo canceled near end signal.

10 518. The data transmission system of claim 507 wherein the signal processor comprises
a voice encoder to process voice including active voice and silent periods, and a voice activity
detector, and wherein the resource manager estimates the signal processor complexity based on
the active voice detection and reduces the signal processor complexity by reducing the
complexity of the voice encoder.

15 519. The data transmission system of claim 507 wherein the signal processor comprises
an decoder, and the resource manager reduces the signal processor complexity with decoder
complexity reductions.

20 520. The data transmission system of claim 519 wherein the decoder includes a post
filter, and the resource manager reduces the decoder complexity by disabling the post filter.

25 521. The data transmission system of claim 507 further comprising a public switched
network coupled between the telephony device and the signal processor.

30 522. The data transmission system of claim 507 wherein the telephony device
comprises a telephone.

35 523. A method of controlling gain applied to a signal, comprising:
 applying gain to the signal;
 estimating a characteristic of the with gain; and
 selectively coupling one of the signal and the signal with gain to an output
depending on the estimated characteristic.

524. The method of claim 523 wherein the characteristic comprises power level.

525. The method of claim 524 wherein the signal is selectively coupled to the output
when the estimated power level of the signal with gain is above a clipping threshold.

526. The method of claim 525 wherein the power level estimation comprises averaging
the power level for a period of time.

1 527. The method of claim 526 wherein the power level estimation further comprises
estimating a second power level by averaging the power level of the signal with gain for a second
period of time longer than the period of time, the method further comprising adjusting the gain
applied to the signal as a function of the second estimated power level.

5 528. The method of claim 527 further comprising peak tracking the second estimated
power level, wherein the gain adjustment is a function of the tracked peak.

 529. The method of claim 528 wherein the gain adjustment comprises changing a rate
of gain adjustment as a function of the tracked peak.

10 530. The method of claim 529 wherein the rate of gain adjustment, when the second
estimated power level is greater than the tracked peak, exceeds the rate of gain adjustment when
the second estimated power level is less than the tracked peak.

15 531. The method of claim 529 wherein the rate of gain adjustment is decreased at about
2-4 dB/sec when a reference value exceeds the clipping threshold, the reference value being a
function of the tracked peak.

20 532. The method of claim 531 wherein the rate of gain adjustment is decreased at
about 0.1-0.3 dB/sec when a reference value is less than the clipping threshold but greater than
a predetermined maximum comfort level, the reference value being a function of the tracked
peak.

25 533. The method of claim 529 wherein the rate of gain adjustment is logarithmically
increased at about 0.1-0.3 dB/sec when a reference value is below a predetermined minimum
comfort level and above a noise floor, the reference value being a function of the tracked peak.

30 534. A method of controlling gain applied to a signal, comprising:
applying gain to the signal;
estimating a characteristic of the signal with gain;
peak tracking the estimated characteristic;
generating a reference value as a function of the tracked peak; and
adjusting the gain applied to the signal as a function of the reference value.

 535. The method of claim 534 wherein the characteristic comprises power level.

35 536. The method of claim 535 wherein the power level estimation comprises averaging
a power level of the signal with gain for a period of time.

 537. The method of claim 536 wherein the power level estimation further comprises
estimating a second power level by averaging the power level of the signal with gain for a second

1 period of time shorter than the period of time, the method further comprising selectively coupling
one of the signal and the signal with gain to an output depending on the second estimated power
level of the signal with gain.

5 538. The method of claim 537 wherein the signal is selectively coupled to the output
when the second estimated power level of the signal with gain is above a clipping threshold.

10 539. The method of claim 535 wherein a rate of change of an amplitude of the
reference value, when the power level is greater than the tracked peak, exceeds the rate of change
of the amplitude of the reference value when the estimated power level is less than the tracked
peak.

540. The method of claim 535 wherein a rate of gain adjustment is decreased at about
2-4 dB/sec when the reference value exceeds a clipping threshold.

15 541. The method of claim 540 wherein the rate of gain adjustment is decreased at
about 0.1-0.3 dB/sec when the reference value is less than the clipping threshold but greater than
a predetermined maximum comfort level.

20 542. The method of claim 535 wherein a rate of gain adjustment is logarithmically
increased at about 0.1-0.3 dB/sec when the reference value is below a predetermined minimum
comfort level and above a noise floor.

25 543. The method of claim 535 wherein the signal with gain comprises first and second
plurality of samples, the first samples preceding the second samples in time, and the reference
value generation comprises not changing the reference value if the estimated power level for the
second samples exceeds the estimated power level for the first samples by a threshold.

30 544. A signal conditioner for adjusting gain applied to a signal, comprising:
a combiner to apply gain to the signal;
an estimator to estimate a characteristic of the signal with gain; and
a bypass to selectively couple one of the signal and the signal with gain to an
output of the signal conditioner based on the estimated characteristic.

545. The signal conditioner of claim 544 wherein the characteristic comprises power
level.

35 546. The signal conditioner of claim 545 wherein the bypass couples the signal with
gain to the output of the signal conditioner when the estimated power level of the signal with gain
is below a clipping threshold.

1 547. The signal conditioner of claim 546 wherein the estimator estimates the power level by averaging the power of the signal for a period of time.

5 548. The signal conditioner of claim 547 wherein the estimator estimates a second power level by averaging the power of the signal for a second period of time longer than the period of time, the signal conditioner further comprising a gain calculator that calculates the gain to be applied to the signal based on the second estimated power level of the signal with gain.

10 549. The signal conditioner of claim 548 further comprising a peak tracker that tracks the second estimated power level peak and outputs a reference value based on the tracked peak, the gain calculator calculating the gain to be applied to the signal based on the reference value.

15 550. The signal conditioner of claim 549 wherein the peak tracker increases an amplitude of the reference value at a first rate when the second estimated power level of the signal with gain is greater than the reference value, and decreases the amplitude of the reference value at a second rate when the second estimated power level of the signal is less than the reference value, the first rate being faster than the second rate.

20 551. The signal conditioner of claim 550 wherein the gain calculator changes a rate of adjustment of the gain applied to the signal as a function of the reference value.

25 552. The signal conditioner of claim 548 wherein the gain calculator decrements the gain applied to the signal at a rate of about 2-4 dB/sec when the reference value exceeds the clipping threshold.

30 553. The signal conditioner of claim 552 wherein the gain calculator decrements the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference value is less than the clipping threshold but greater than a predetermined maximum comfort level.

35 554. The signal conditioner of claim 551 wherein the gain calculator logarithmically increases the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference value is below a predetermined minimum comfort level and above a noise floor.

55 555. A signal conditioner for adjusting gain applied to a signal, comprising:
a combiner to apply gain to the signal;
an estimator which estimates a characteristic of the signal with gain;
a peak tracker that tracks the estimated characteristic peak and generates a
reference value as a function of the tracked peak; and
a gain calculator that calculates the gain to be applied to the signal as a function
of the reference value.

1 556. The signal conditioner of claim 555 wherein the characteristic comprises power level.

5 557. The signal conditioner of claim 556 wherein the estimator estimates the power level by averaging the power level of the signal with gain over a period of time.

10 558. The signal conditioner of claim 557 wherein the estimator estimates a second power level by averaging the power level of the signal with gain over a second period of time shorter than the period of time, the signal conditioner further comprising a bypass to selectively couple one of the signal and the signal with gain to an output of the signal conditioner as a function of the second estimated power level of the signal with gain.

15 559. The signal conditioner of claim 558 wherein the bypass couples the signal with gain to the output of the signal conditioner when the second estimated power level of the signal with is below a clipping threshold.

20 560. The signal conditioner of claim 556 wherein the peak tracker increases an amplitude of the reference value at a first rate when the estimated power level of the signal with gain is greater than the reference value, and decreases the amplitude of the reference value at a second rate when the estimated power level of the signal with gain is less than the reference value, the first rate being faster than the second rate.

25 561. The signal conditioner of claim 555 wherein the gain calculator changes a rate of gain adjustment as a function of the reference value.

30 562. The signal conditioner of claim 561 wherein the gain calculator decrements the gain applied to the signal at a rate of about 2-4 dB/sec when the reference value exceeds the clipping threshold.

35 563. The signal conditioner of claim 562 wherein the gain calculator decrements the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference value is less than the clipping threshold but greater than a predetermined maximum comfort level.

 564. The signal conditioner of claim 561 wherein the gain calculator logarithmically increases the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference value is below a predetermined minimum comfort level and above a noise floor.

 565. The signal conditioner of claim 555 wherein the combiner comprises a multiplier.

 566. A data transmission system, comprising:
 a telephony device which outputs a signal; and

1 a signal processor comprising a combiner to apply gain to the signal, an estimator to estimate a characteristic of the signal with gain, and a bypass to selectively couple one of the signal and the signal with gain to an output of the signal processor based on the estimated characteristic.

5 567. The data transmission system of claim 566 wherein the characteristic comprises power level.

10 568. The data transmission system of claim 567 wherein the bypass couples the signal with gain to the output of the signal processor when the estimated power level of the signal with gain is below a clipping threshold.

 569. The data transmission system of claim 568 wherein the estimator estimates the power level by averaging the power level of the signal for a period of time.

15 570. The data transmission system of claim 569 wherein the estimator estimates a second power level by averaging the power level of the signal for a second period of time longer than the period of time, the signal processor further comprises a gain calculator that calculates the gain to be applied to the signal based on the second estimated power level of the signal with gain.

20 571. The data transmission system of claim 470 wherein the signal processor further comprises a peak tracker that tracks the second estimated power level peak and outputs a reference value based on the tracked peak, the gain calculator calculating the gain to be applied to the signal based on the reference value.

25 572. The data transmission system of claim 471 wherein the peak tracker increases an amplitude of the reference value at a first rate when the second estimated power level of the signal with gain is greater than the reference value, and decreases the amplitude of the reference value at a second rate when the second estimated power level of the signal is less than the reference value, the first rate being faster than the second rate.

30 573. The data transmission system of claim 571 wherein the gain calculator changes a rate of adjustment of the gain applied to the signal as a function of the reference value.

35 574. The data transmission system of claim 573 wherein the gain calculator decrements the gain applied to the signal at a rate of about 2-4 dB/sec when the reference value exceeds the clipping threshold.

 575. The data transmission system of claim 574 wherein the gain calculator decrements the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference

1 value is less than the clipping threshold but greater than a predetermined maximum comfort level.

5 576. The data transmission system of claim 573 wherein the gain calculator logarithmically increases the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference value is below a predetermined minimum comfort level and above a noise floor.

577. The data transmission system of claim 566 wherein the telephony device comprises a telephone.

10 578. The data transmission system of claim 566 further comprising a public switched telephone network coupled between the telephony device and the signal processor.

579. A data transmission system, comprising:
a telephony device which outputs a signal; and
15 a signal processor comprising a combiner to apply gain to the signal, an estimator which estimates a characteristic of the signal with gain, a peak tracker that tracks the estimated characteristic peak and generates a reference value as a function of the tracked peak, and a gain calculator that calculates the gain to be applied to the signal as a function of the reference value.

20 580. The data transmission system of claim 579 wherein the characteristic comprises power level.

581. The data transmission system of claim 580 wherein the estimator estimates the power level by averaging the power level of the signal with gain over a period of time.

25 582. The data transmission system of claim 581 wherein the estimator estimates a second power level by averaging the power level of the signal with gain over a second period of time shorter than the period of time, and wherein the signal processor further comprises a bypass to selectively couple one of the signal and the signal with gain to an output of the signal processor as a function of the second estimated power level of the signal with gain.

30 583. The data transmission system of claim 582 wherein the bypass couples the signal with gain to the output of the signal processor when the second estimated power level of the signal with is below a clipping threshold.

35 584. The data transmission system of claim 580 wherein the peak tracker increases an amplitude of the reference value at a first rate when the estimated power level of the signal with gain is greater than the reference value, and decreases the amplitude of the reference value at a second rate when the estimated power level of the signal with gain is less than the reference value, the first rate being faster than the second rate.

1 585. The data transmission system of claim 579 wherein the gain calculator changes
a rate of gain adjustment as a function of the reference value.

5 586. The data transmission system of claim 585 wherein the gain calculator decrements
the gain applied to the signal at a rate of about 2-4 dB/sec when the reference value exceeds the
clipping threshold.

10 587. The data transmission system of claim 586 wherein the gain calculator
decrements the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the reference
value is less than the clipping threshold but greater than a predetermined maximum comfort
level.

15 588. The data transmission system of claim 585 wherein the gain calculator
logarithmically increases the gain applied to the signal at a rate of about 0.1-0.3 dB/sec when the
reference value is below a predetermined minimum comfort level and above a noise floor.

20 589. The data transmission system of claim 579 wherein the telephony device
comprises a telephone.

25 590. The data transmission system of claim 579 further comprising a public switched
telephone network coupled between the telephony device and the signal processor.

30 591. The data transmission system of claim 579 wherein the combiner comprises a
multiplier.

35 592. A method of detecting a call progress tone in a signal, comprising:
selectively analyzing spectral content of the signal;
generating an indicator if the analyzed spectral content of the signal satisfies a
criteria;
monitoring a temporal characteristic of the indicator; and
detecting the call progress tone based on the monitored temporal characteristic.

40 593. The method of claim 592 further comprising filtering the signal before analyzing
the spectral characteristic.

45 594. The method of claim 593 wherein the filtering comprises removing frequency
components in the signal above a threshold.

50 595. The method of claim 593 further comprising downsampling the filtered signal
before analyzing the spectral characteristic.

1 596. The method of claim 595 further comprising estimating power of the
downsampled signal, comparing the estimated power with at least one threshold, and invoking
the spectral content analysis based on the comparison.

5 597. The method of claim 596 wherein the comparison of the estimated power with
said at least one threshold comprises generating a series of power indicators over time, the
spectral content analysis being invoked upon the generation of consecutive power indicators
each satisfying a power criteria.

10 598. The method of claim 597 wherein the spectral content analysis comprises
differentially detecting a frequency of the downsampled signal.

 599. The method of claim 598 wherein the spectral content analysis comprises
estimating a mean the estimated frequency, and comparing the estimated mean to a frequency
range.

15 600. The method of claim 592 wherein the monitoring of the temporal characteristic
of the indicator comprises estimating a duration of the indicator over time, and comparing the
estimated duration to a threshold, the call progress tone detection being based on the comparison.

20 601. A method for detecting a call progress tone in a composite signal having a
plurality of components, comprising:
 separating the components of the composite signal;
 analyzing spectral content for each of the separated components;
 selectively generating an indicator for each of the separated components whose
spectral content satisfies a respective criteria;
25 monitoring a temporal characteristic for each of the indicators; and
 detecting the call progress tone in the composite signal based on the monitored
temporal characteristics.

 602. The method of claim 601 further comprising filtering the composite signal.

30 603. The method of claim 602 wherein the filtering of the composite signal comprises
removing frequency components in the composite signal above a threshold.

 604. The method of claim 602 further comprising downsampling the filtered composite
signal.

35 605. The method of claim 604 wherein the separation of the components of the
composite signal comprises bandpass filtering the downsampled composite signal.

1 606. The method of claim 605 wherein the bandpass filtering of the downsampled composite signal comprises converting each of the separated components into a complex component.

5 607. The method of claim 601 further comprising estimating power for each of the components, comparing the estimated power for each of the components with a respective threshold, and invoking the spectral content analysis for each component based on the comparison.

10 608. The method of claim 607 wherein the comparison comprises generating a series of power indicators over time for each component, the spectral content analysis for each component being invoked upon the generation of respective consecutive power indicators each satisfying a power criteria.

15 609. The method of claim 608 wherein the spectral content analysis for each of the components comprises differentially detecting a frequency of each for the components.

 610. The method of claim 609 wherein the spectral content analysis comprises estimating a mean of the estimated frequency for each of the components, and comparing the estimated mean to a respective frequency range.

20 611. The method of claim 601 wherein the monitoring of the temporal characteristic for each of the indications comprises estimating a duration of the respective indicator over time, and comparing the estimated duration for each of the components to at least one respective threshold, the call progress tone detection being based on the comparison.

25 612. A system for detecting a call progress tone in a signal, comprising:
 a signal processor to selectively analyze spectral content of the signal and generate an indicator if the spectral content of the signal satisfies a criteria; and
 a cadence processor to monitor a temporal characteristic of the indicator and detect the call progress tone based on the temporal characteristic.

30 613. The system of claim 612 wherein the signal processor comprises a low pass filter to filter the signal, and a downsampler to decimate the filtered signal, the decimated signal being analyzed for spectral content.

35 614. The system of claim 613 wherein the call progress tone comprises one of a plurality of tones each having a frequency, and wherein the low pass filter removes frequency components in the signal above the highest frequency.

 615. The system of claim 612 wherein the signal processor comprises a differential detector to analyze the spectral content of the signal by estimating a frequency for the signal.

1 616. The system of claim 615 wherein the signal processor further comprising a power estimator to estimate power of the signal and to generate a power indicator based on the power estimation, and a power state machine to enable the differential detector based on the power indicator.

5 617. The system of claim 612 wherein the cadence processor comprises a cadence state machine responsive to the indicator, a counter enabled by the cadence state machine and which estimates cadence of the indicator, and cadence logic to compare the cadence of the indicator to a threshold to detect the call progress tone.

10 618. A system for detecting a call progress tone in a composite signal having a plurality of components, comprising:

 a plurality of bandpass filters to separate the components of the composite signal;
 a plurality of differential detectors each which estimates a frequency for one of the components;

15 a plurality of frequency calculators each which analysis a mean of the estimated frequency for one of the components and generates a tone indicator as a function of the analysis; and

 a cadence processor that monitors a temporal characteristic of each of the tone indicators and detects the call progress tone in the composite signal based on the temporal monitoring.

20 619. The system of claim 618 further comprising a downsampler to decimate the composite signal before the composite signal is separated into its components by the bandpass filters.

25 620. The system of claim 618 wherein the bandpass filters each comprises a complex filter.

30 621. The system of claim 618 further comprising a plurality of power estimators each which estimate power for one of the components and generates a power indicator as a function of the estimation, and a plurality of power state machines each which monitors the power indicator for one of the components and invokes a respective one of the frequency calculators in response thereto.

35 622. The system of claim 618 wherein the cadence processor comprises a cadence state machine responsive to the tone indicators, a counter to estimate cadence of the tone indicators, and cadence logic which compares the cadence of the tone indicators to a respective threshold to detect the call progress tone in the composite signal.

623. A data transmission system, comprising:
 a telephony device which outputs a signal having a call progress tone; and

1 a data exchange coupled to the telephony device, the data exchange comprising
a signal processor to selectively analyze spectral content of the signal and generate an indicator
if the spectral content of the signal satisfies a criteria, and a cadence processor to monitor a
temporal characteristic of the indicator and detect the call progress tone based on the temporal
5 characteristic.

624. The data transmission system of claim 623 wherein the signal processor comprises
a low pass filter to filter the signal, and a downsampler to decimate the filtered signal, the
decimated signal being analyzed for spectral content.

10 625. The data transmission system of claim 624 wherein the call progress tone
comprises one of a plurality of tones each having a frequency, and wherein the low pass filter
removes frequency components in the signal above the highest frequency.

15 626. The data transmission system of claim 623 wherein the signal processor comprises
a differential detector to analyze the spectral content of the signal by estimating a frequency for
the signal.

20 627. The system of claim 626 wherein the signal processor further comprising a power
estimator to estimate power of the signal and to generate a power indicator based on the power
estimation, and a power state machine to enable the differential detector based on the power
indicator.

25 628. The system of claim 623 wherein the cadence processor comprises a cadence state
machine responsive to the indicator, a counter enabled by the cadence state machine and which
estimates cadence of the indicator, and cadence logic to compare the cadence of the indicator to
a threshold to detect the call progress tone.

629. The data transmission system of claim 623 wherein the telephony device
comprises a telephone.

30 630. The data transmission system of claim 623 further comprising a public switched
telephone network coupling the telephony device to the data exchange.

35 631. A method of detecting voice in a signal, comprising:
autocorrelating the signal;
estimating a characteristic of the autocorrelated signal; and
detecting voice in the signal as a function of the estimated characteristic.

632. The method of claim 631 wherein the signal comprises first, second and third
frames, the first frame preceding the second frame in time and the second frame preceding the

1 third frame in time, the method further comprising vacating the voice detection for the second
frame if voice is not detected in both the first and third frames.

5 633. The method of claim 631 further comprising estimating power of the signal, and
comparing the estimated power of the signal to a power threshold, the detection of voice in the
signal being further a function of the estimated power comparison.

634. The method of claim 633 wherein the power threshold is in the range of -45 to -55
dBm.

10 635. The method of claim 631 wherein the characteristic comprises pitch period.

636. The method of claim 635 wherein the detection of voice in the signal is further
based on the estimated pitch period of the autocorrelated signal being in the range of 60-400 Hz.

15 637. The method of claim 636 wherein the characteristic comprises amplitude, and the
voice detection comprises detecting the amplitude of the autocorrelated signal with one period
shift and with no shift, the voice detection being further based on the amplitude of autocorrelated
signal with one period shift being in the range of 0.25-0.40 of the amplitude of the autocorrelated
signal with no shift.

20 638. The method of claim 636 wherein the characteristic comprises peak amplitude,
and the voice detection comprises detecting the peak amplitude of the autocorrelated signal with
no shift and with a shift, the detection of voice in the signal being further based on the peak
amplitude of the shifted autocorrelated signal being less than 0.75 to 0.90 of the peak amplitude
of the autocorrelated signal with no shift.

25 639. A voice detector, comprising:
autocorrelation logic to autocorrelate a signal; and
frame based decision logic that detects voice in the signal as a function of the
autocorrelated signal.

30 640. The voice detector of claim 639 wherein the signal comprises first, second and
third frames, the first frame preceding the second frame in time and the second frame preceding
the third frame in time, the voice detector further comprising final decision logic which vacates
the detection of voice in the signal for the second frame if voice is not detected by the frame
based decision logic for both the first and third frames.

35 641. The voice detector of claim 639 further comprising a pitch period tracker to
estimate a pitch period of the autocorrelated, and wherein the frame based decision logic detects
voice in the signal as a function of the estimated pitch period of the autocorrelated signal.

1 642. The voice detector of claim 641 further comprising a power estimator which estimates power of the signal, and wherein the frame based decision logic further compares the estimated power of the signal to a power threshold, the detection of voice in the signal being further a function of the power comparison.

5 643. The voice detector of claim 642 wherein the power threshold is in the range of -45 to -55 dBm, and the detection of voice in the signal is further based on the estimated power exceeding the power threshold.

10 644. The voice detector of claim 642 wherein the detection of voice in the signal by the frame based decision logic is further based on the estimated pitch period for the autocorrelated signal being in the range of 60-400 Hz.

15 645. The voice detector of claim 644 wherein the frame based decision logic further detects an amplitude for the autocorrelated signal with one period shift and with no shift, the detection of voice in the signal being further based on the amplitude of the autocorrelated signal with one period being in the range of 0.25-0.40 of the amplitude of the autocorrelated signal with no shift.

20 646. The voice detector of claim 644 wherein the frame based decision logic further detects a peak amplitude of the autocorrelated signal with no shift and with a shift, the detection of voice in the signal being further based on the peak amplitude of the shifted autocorrelated signal being less than 0.75 to 0.90 of the peak amplitude of the autocorrelated signal with no shift.

25 647. A transmission system, comprising:
a telephony device which outputs a signal; and
a voice detector having autocorrelation logic to autocorrelate the signal, and frame based decision logic that detects voice in the signal as a function of the autocorrelated signal .

30 648. The transmission system of claim 647 wherein the signal comprises first, second and third frames, the first frame preceding the second frame in time and the second frame preceding the third frame in time, the voice detector further comprising final decision logic which vacates the detection of voice in the signal for the second frame if voice is not detected by the frame based decision logic for both the first and third frames.

35 649. The transmission system of claim 647 wherein the voice detector further comprises a pitch period tracker to estimate a pitch period of the autocorrelated, and wherein the frame based decision logic detects voice in the signal as a function of the estimated pitch period of the autocorrelated signal.

1 650. The transmission system of claim 649 wherein the voice detector further comprises a power estimator which estimates power of the signal, and wherein the frame based decision logic further compares the estimated power of the signal to a power threshold, the detection of voice in the signal being further a function of the power comparison.

5 651. The transmission system of claim 650 wherein the power threshold is in the range of -45 to -55 dBm, and the detection of voice in the signal is further based on the estimated power exceeding the power threshold.

10 652. The transmission system of claim 649 wherein the detection of voice in the signal by the frame based decision logic is further based on the estimated pitch period for the autocorrelated signal being in the range of 60-400 Hz.

15 653. The transmission system of claim 652 wherein the frame based decision logic further detects an amplitude for the autocorrelated signal with one period shift and with no shift, the detection of voice in the signal being further based on the amplitude of the autocorrelated signal with one period being in the range of 0.25-0.40 of the amplitude of the autocorrelated signal with no shift.

20 654. The transmission system of claim 652 wherein the frame based decision logic further detects a peak amplitude of the autocorrelated signal with no shift and with a shift, the detection of voice in the signal being further based on the peak amplitude of the shifted autocorrelated signal being less than 0.75 to 0.90 of the peak amplitude of the autocorrelated signal with no shift.

25 655. The transmission system of claim 647 wherein the telephony device comprises a telephone.

 656. The transmission system of claim 647 further comprising a public switched telephone network coupling the telephony device to the voice detector.

30 657. A system for processing a signal, comprising:
 a voice exchange capable of exchanging voice in the signal between a telephony device and a network;
 a voiceband data exchange capable of exchanging data in the signal between a data device and the network;
 a voice detector to detect voice in the signal during the voice band data exchange
35 ; and
 a resource manager which terminates the voiceband data exchange and invokes the voice exchange when the voice detector detects voice in the signal.

1 658. The signal processing system of claim 657 wherein the voice detector comprises autocorrelation logic to autocorrelate the signal, and frame based decision logic that detects voice in the signal as a function of the autocorrelated signal.

5 659. The signal processing system of claim 658 wherein the signal comprises first, second and third frames, the first frame preceding the second frame in time and the second frame preceding the third frame in time, the voice detector further comprising final decision logic which vacates the detection of voice in the signal for the second frame if voice is not detected by the frame based decision logic for both the first and third frames.

10 660. The signal processing system of claim 658 wherein the voice detector further comprises a pitch period tracker to estimate a pitch period of the autocorrelated, and wherein the frame based decision logic detects voice in the signal as a function of the estimated pitch period of the autocorrelated signal to a threshold.

15 661. The signal processing system of claim 660 wherein the voice detector further comprises a power estimator which estimates power of the signal, and wherein the frame based decision logic further compares the estimated power of the signal to a power threshold, the detection of voice in the signal being further a function of the power comparison.

20 662. The signal processing system of claim 661 wherein the power threshold is in the range of -45 to -55 dBm, and the detection of voice in the signal is further based on the estimated power exceeding the power threshold.

25 663. The signal processing system of claim 660 wherein the detection of voice in the signal by the frame based decision logic is further based on the estimated pitch period for the autocorrelated signal being in the range of 60-400 Hz.

30 664. The signal processing system of claim 663 wherein the frame based decision logic further detects an amplitude for the autocorrelated signal with one period shift and with no shift, the detection of voice in the signal being further based on the amplitude of the autocorrelated signal with one period being in the range of 0.25-0.40 of the amplitude of the autocorrelated signal with no shift.

35 665. The signal processing system of claim 663 wherein the frame based decision logic further detects a peak amplitude of the autocorrelated signal with no shift and with a shift, the detection of voice in the signal being further based on the peak amplitude of the shifted autocorrelated signal being less than 0.75 to 0.90 of the peak amplitude of the autocorrelated signal with no shift.

666. A method of processing a signal, comprising:

1 invoking a data exchange service to exchange data in the signal between a data device and a network;

 invoking a voice detection service to detect voice in the signal when the data exchange service is invoked; and

5 terminating the data exchange service and invoking a voice exchange service when the voice detector detects voice in the signal.

667. The method of claim 666 wherein the invoked voice detection service comprises autocorrelating the signal, estimating a characteristic of the autocorrelated signal; and detecting voice in the signal as a function of the estimated characteristic.

10 668. The method of claim 667 wherein the signal comprises first, second and third frames, the first frame preceding the second frame in time and the second frame preceding the third frame in time, the voice detector further comprising vacating the detection of voice in the signal for the second frame if voice is not detected by the frame based decision logic for both the first and third frames.

15 669. The method of claim 667 wherein the invoked voice detection service further comprising estimating power of the signal, and comparing the estimated power of the signal to a power threshold, the detection of voice in the signal being further a function of the estimated power comparison.

20 670. The method of claim 669 wherein the power threshold is in the range of -45 to -55 dBm.

 671. The method of claim 667 wherein the characteristic comprises pitch period.

25 672. The method of claim 671 wherein the detection of voice in the signal is further based on an autocorrelation pitch period in the range of 60-400 Hz.

30 673. The method of claim 672 wherein the characteristic comprises amplitude, and the invoked voice detection service further comprises detecting the amplitude of the autocorrelated signal with one period shift and with no shift, the detection of voice in the signal being further based on the amplitude of autocorrelated signal with one period shift being in the range of 0.25-0.40 of the amplitude of the autocorrelated signal with no shift.

35 674. The method of claim 672 wherein the characteristic comprises peak amplitude, and the invoked voice detection service comprises detecting the peak amplitude of the autocorrelated signal with no shift and with a shift, the detection of voice in the signal being further based on the peak amplitude of the shifted autocorrelated signal being less than 0.75 to 0.90 of the peak amplitude of the autocorrelated signal with no shift.

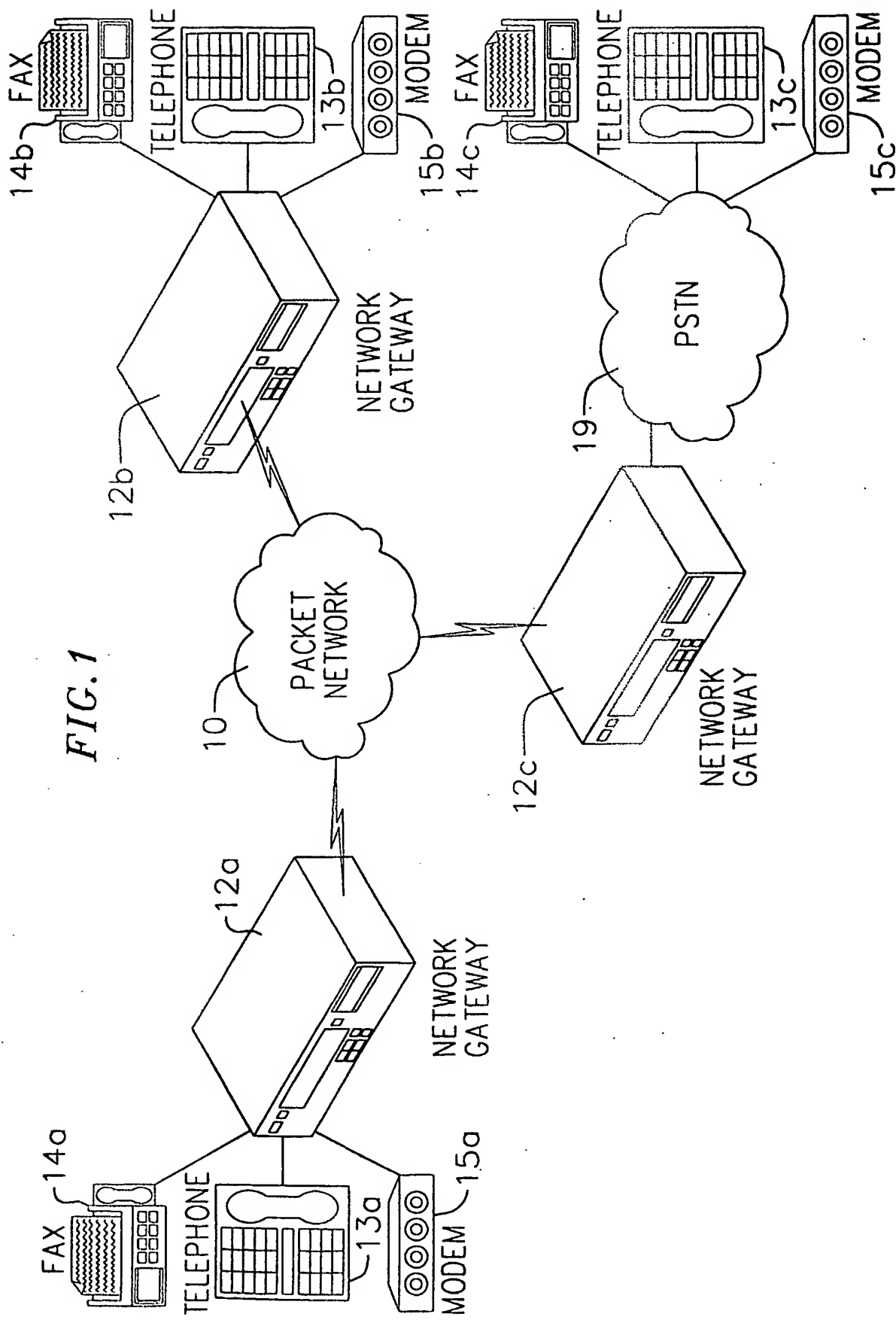


FIG. 1

FIG. 2

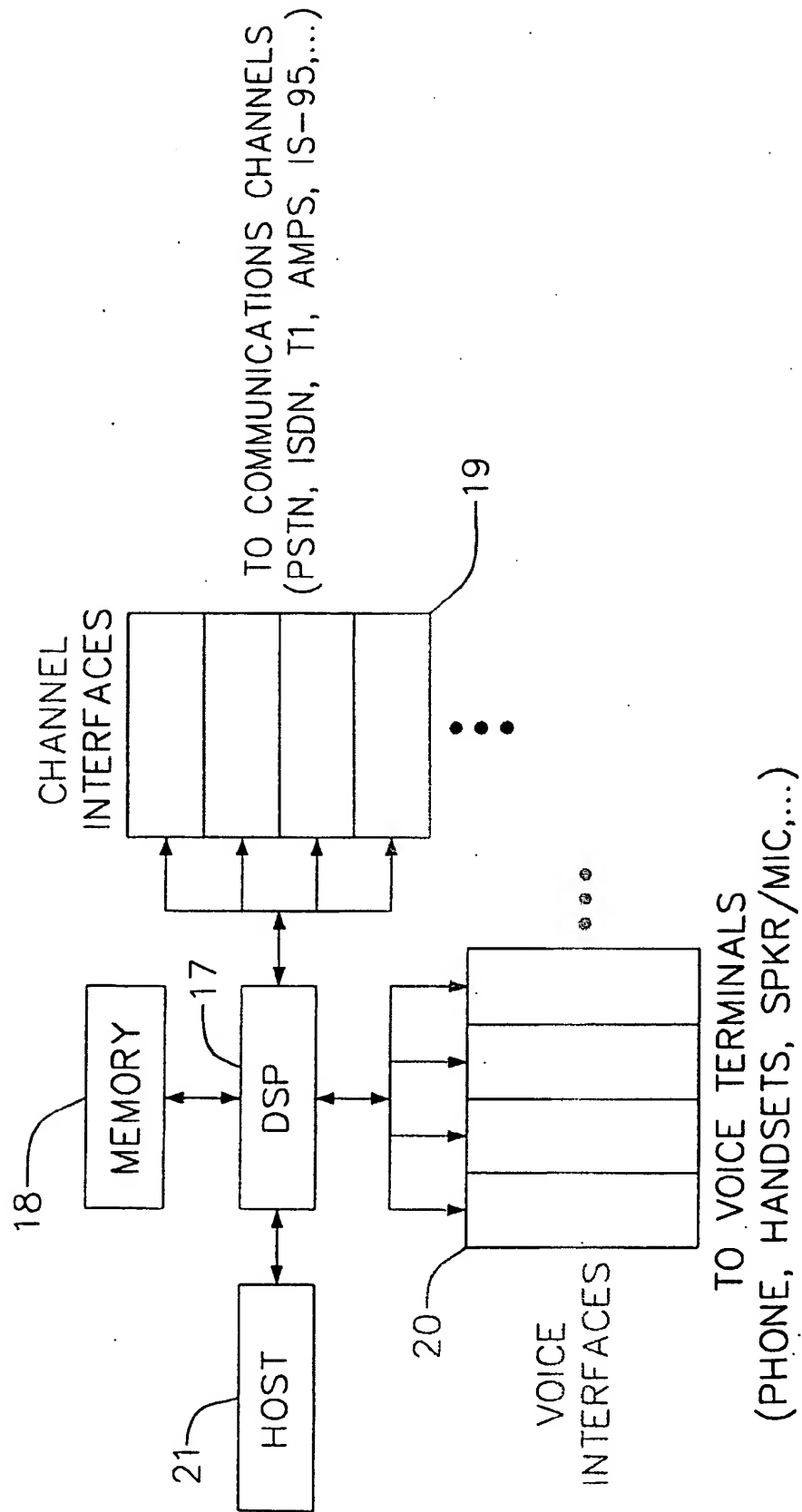


FIG.3

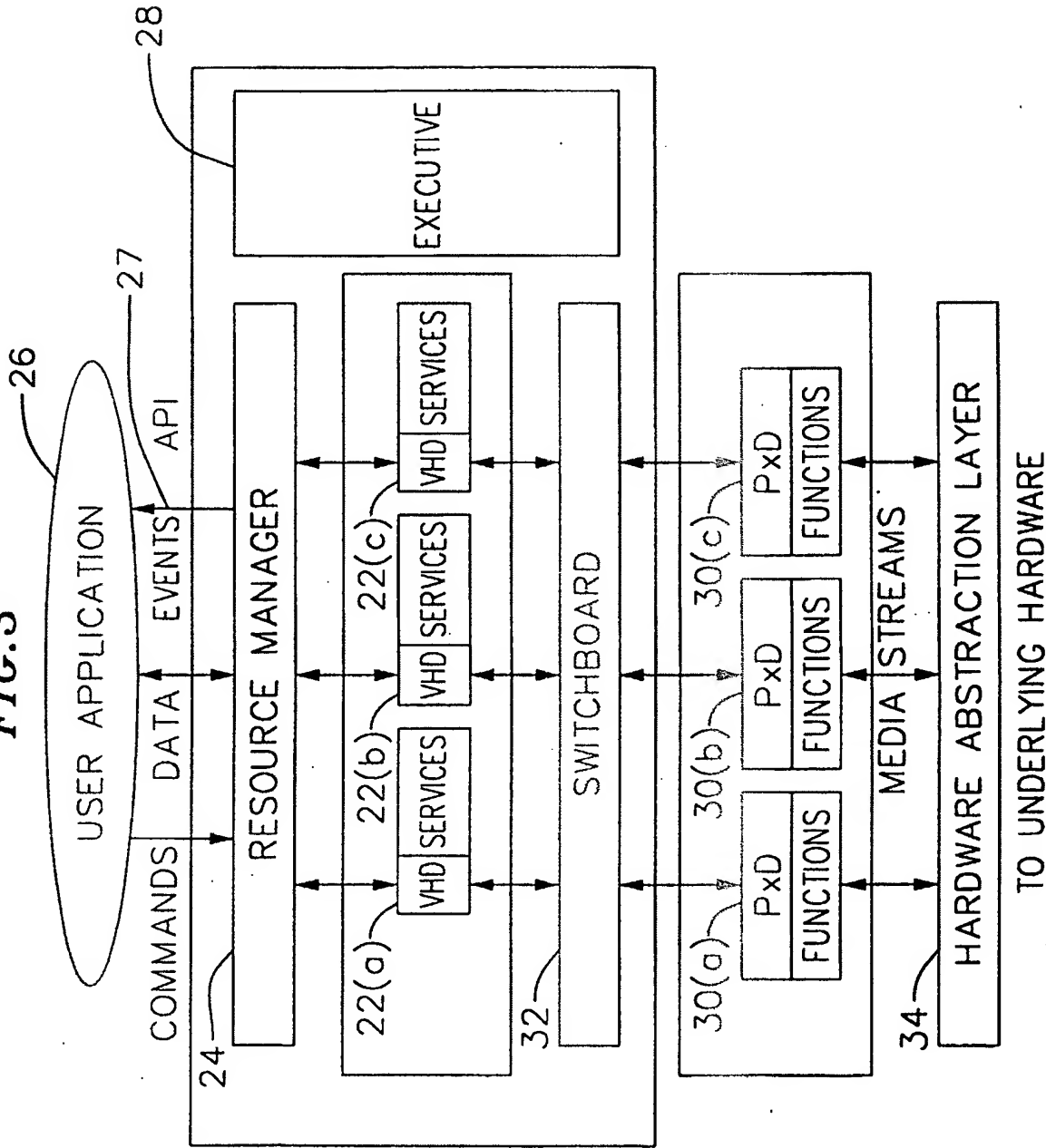


FIG. 4

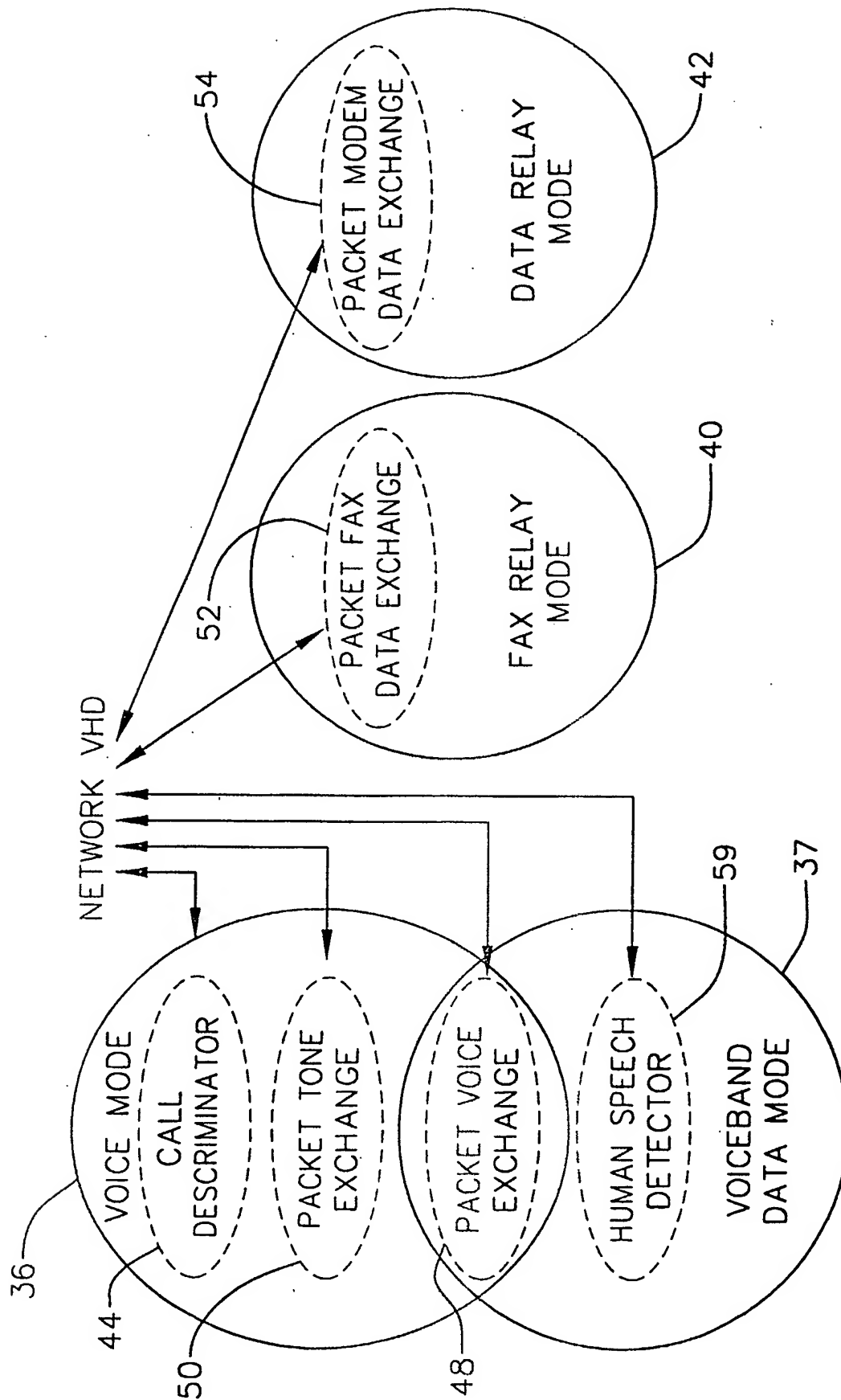


FIG. 5

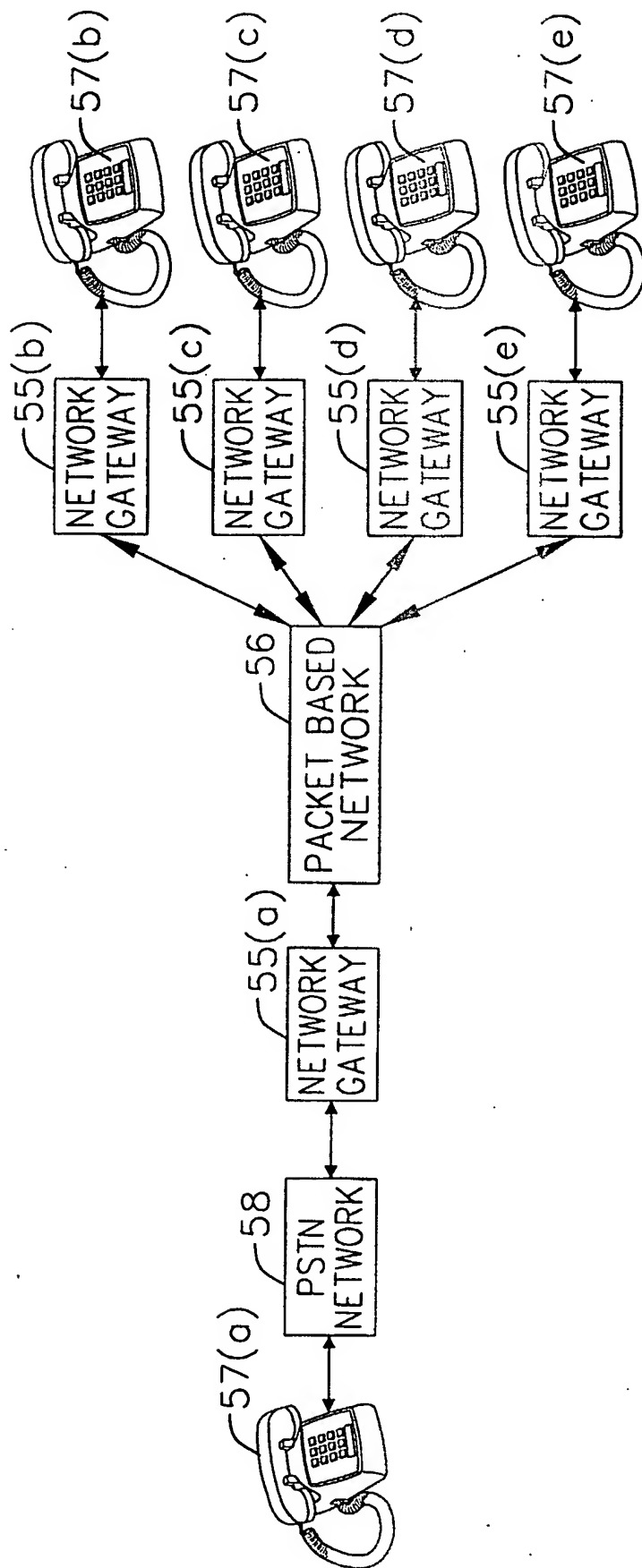


FIG. 6

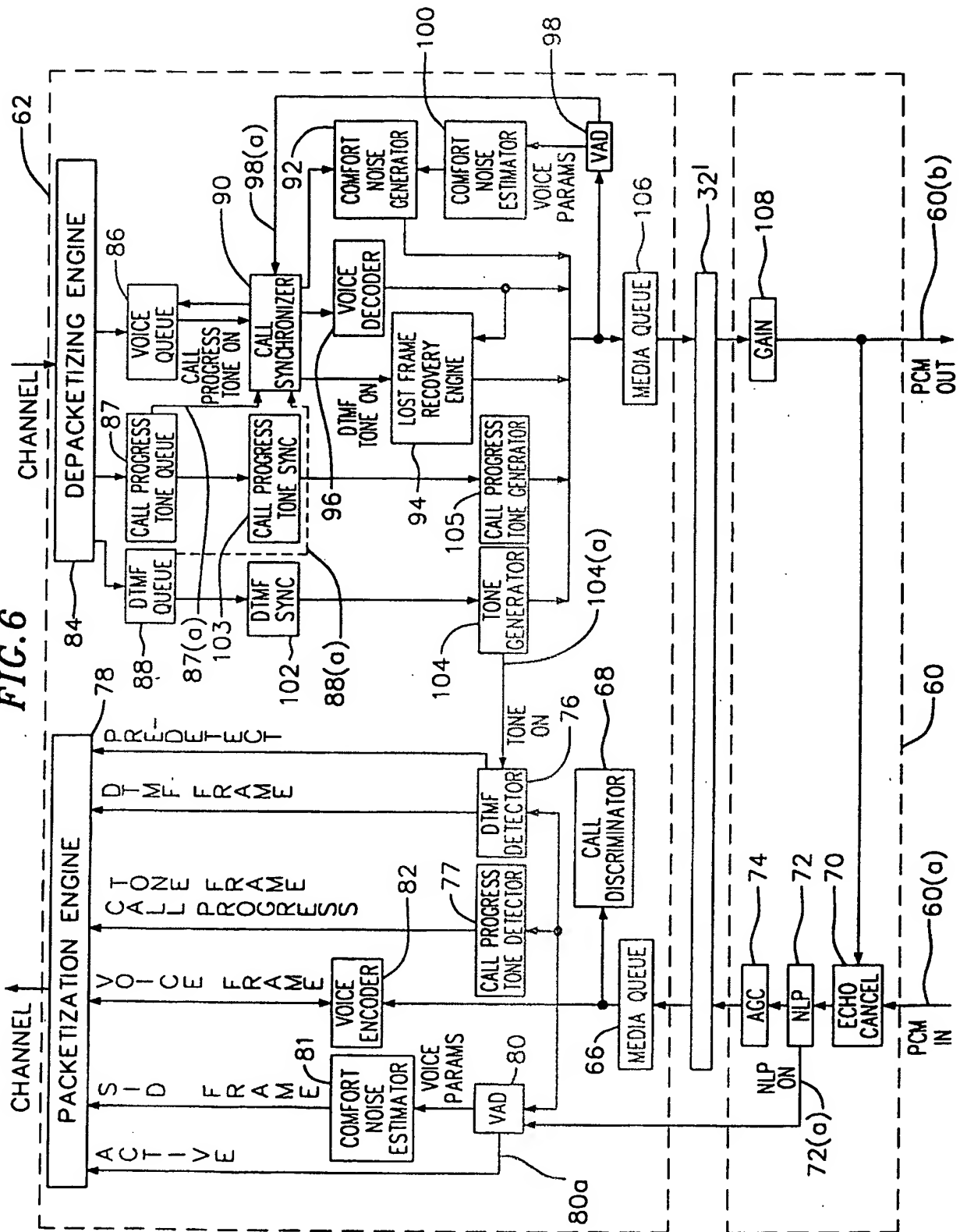


FIG. 8A

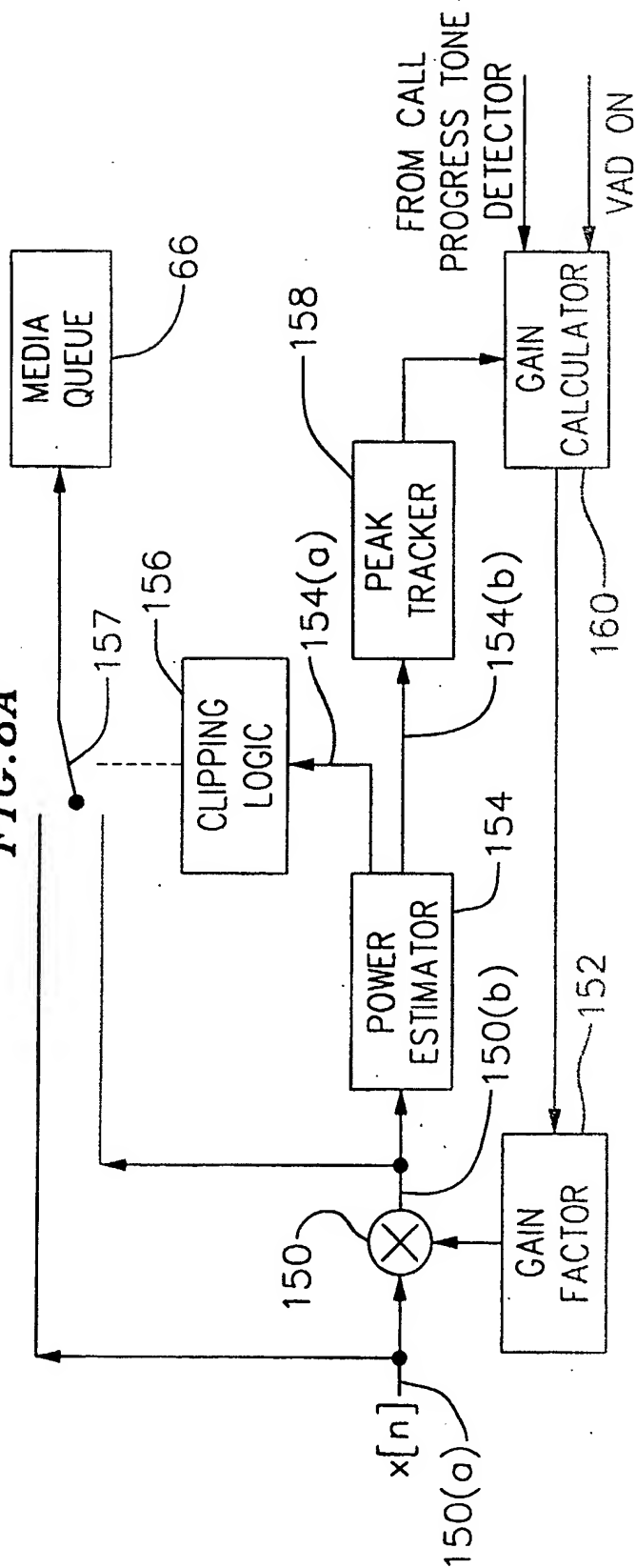


FIG. 8B

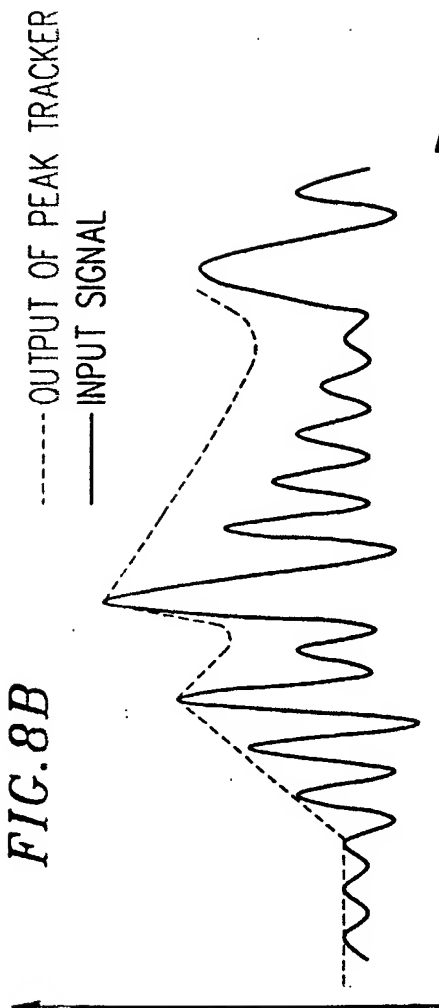
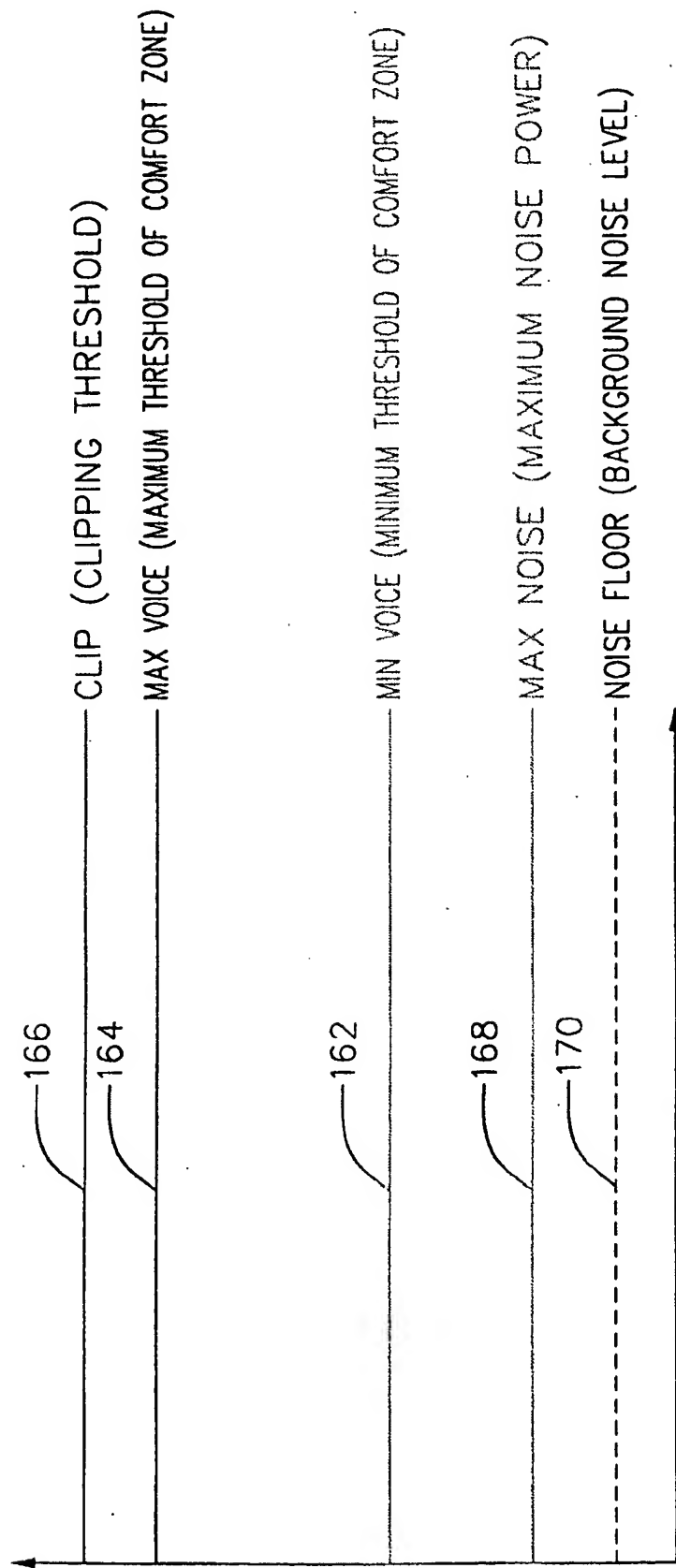


FIG. 9



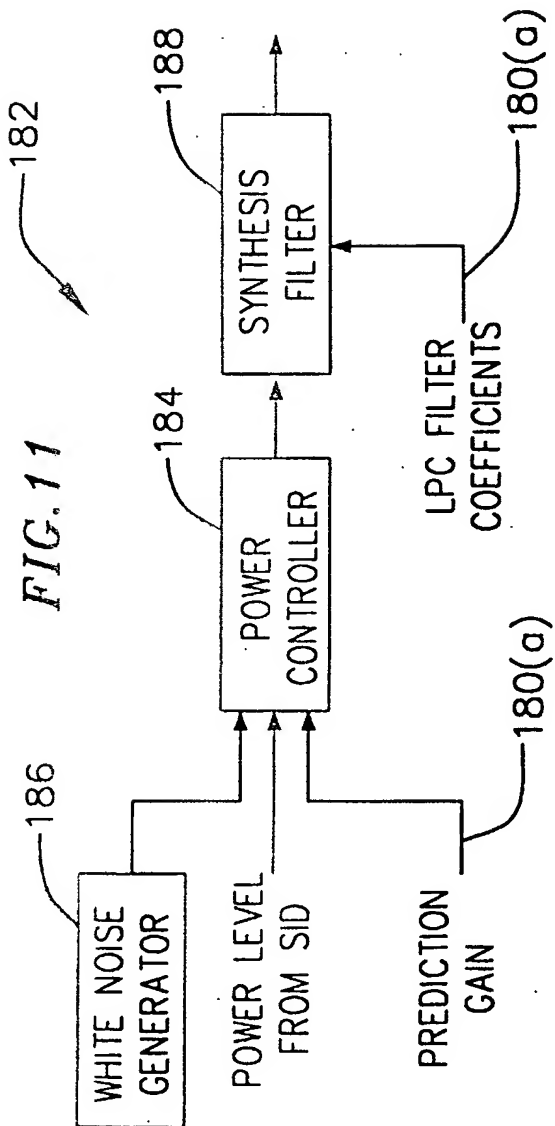
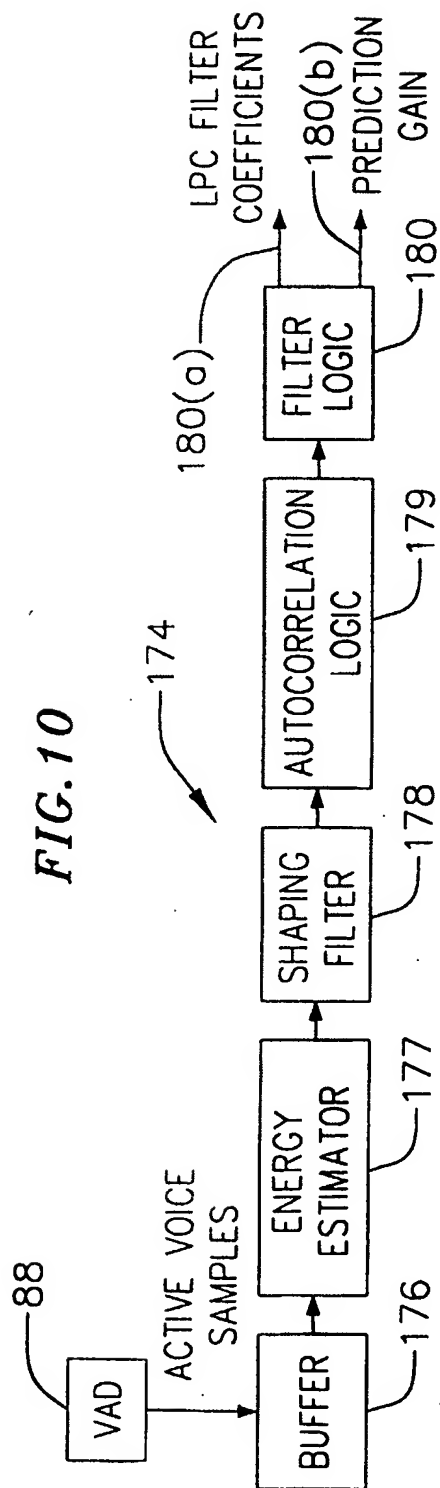


FIG. 12

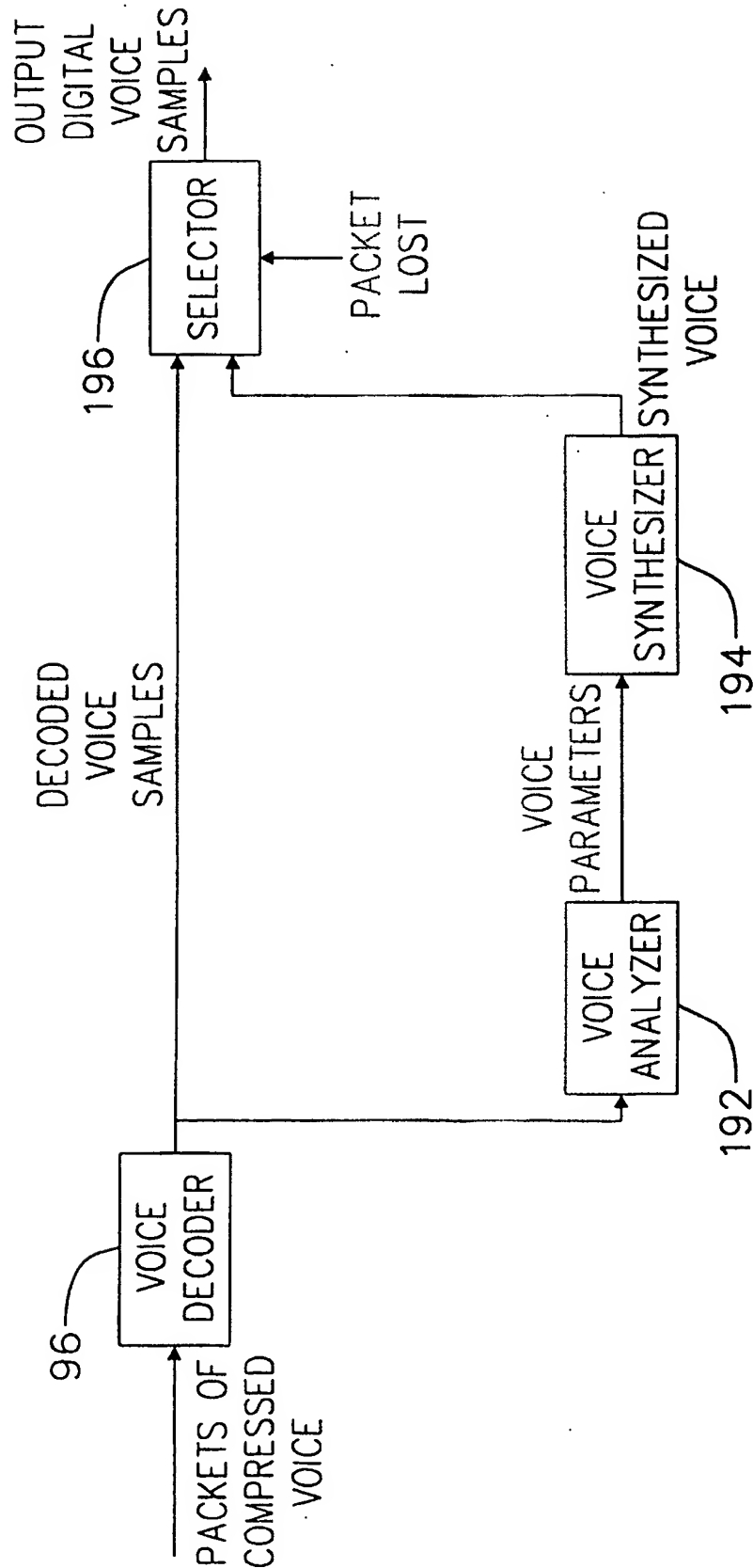


FIG. 13A

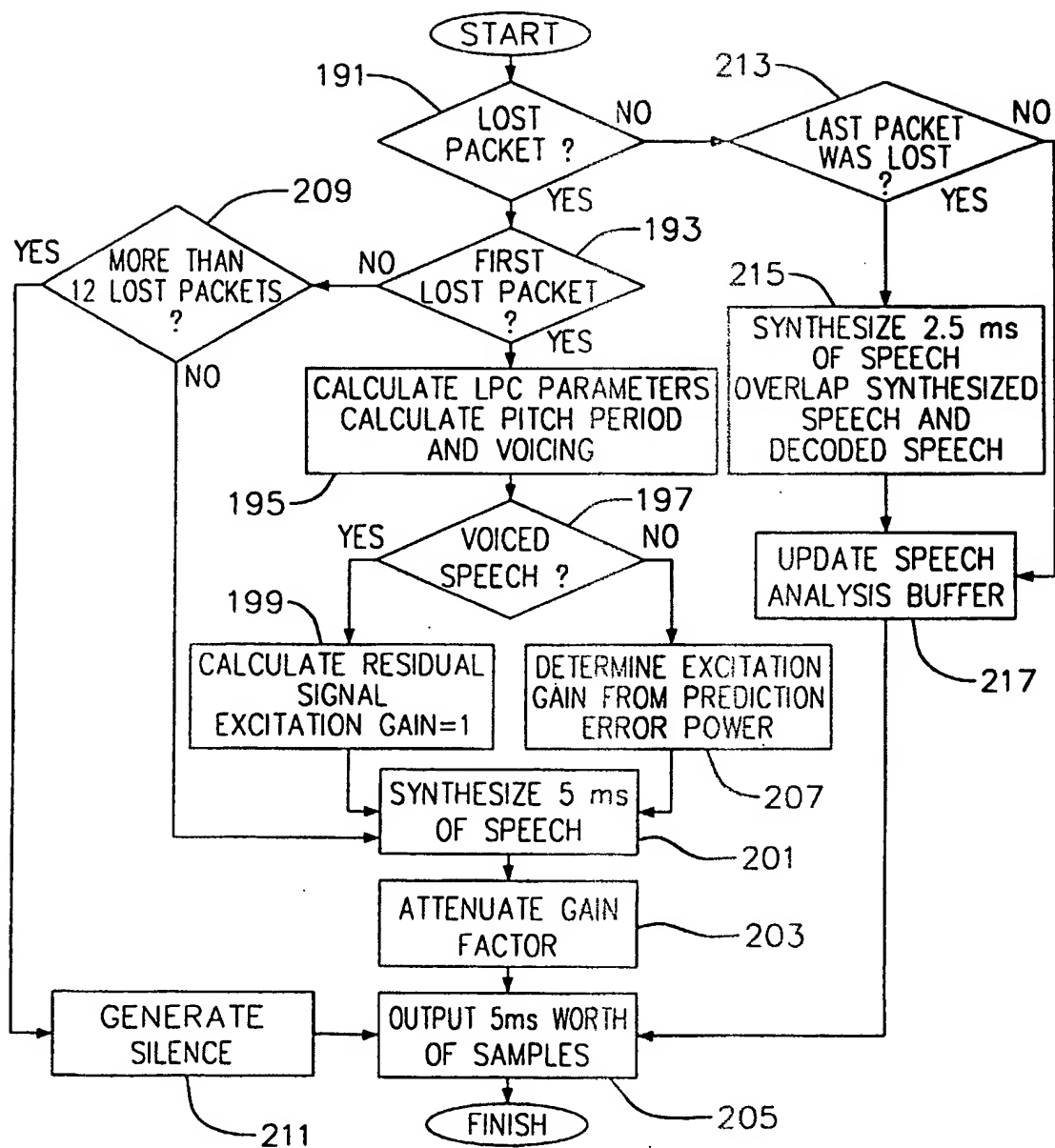


FIG. 13B

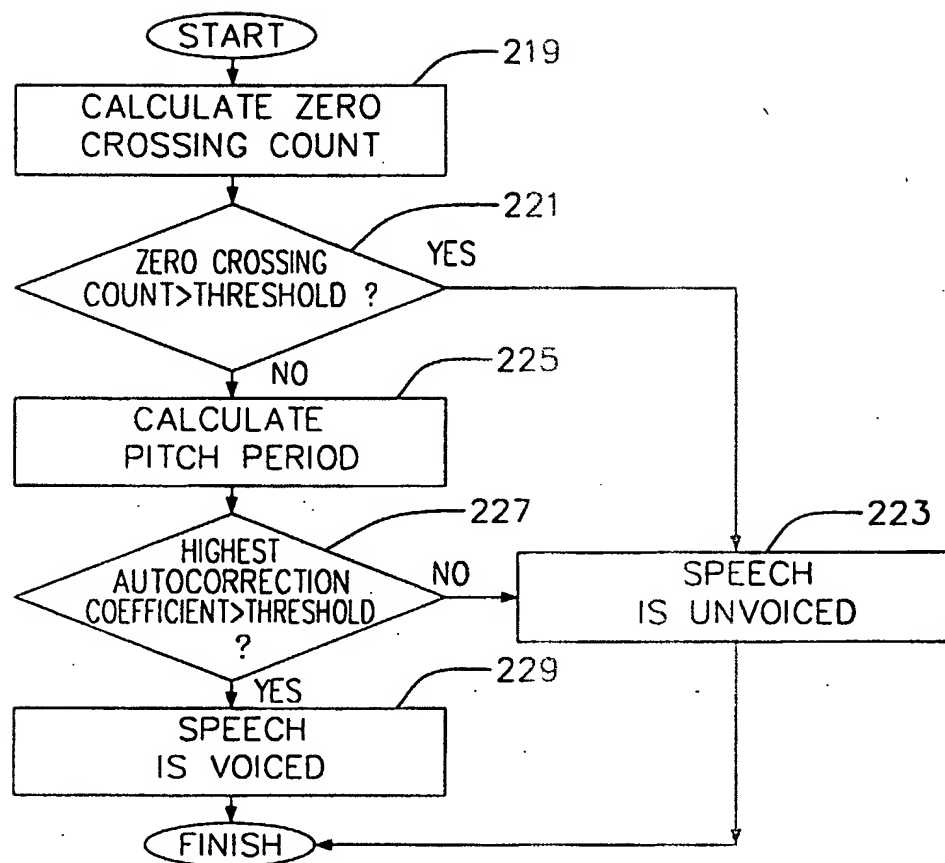


FIG. 13C

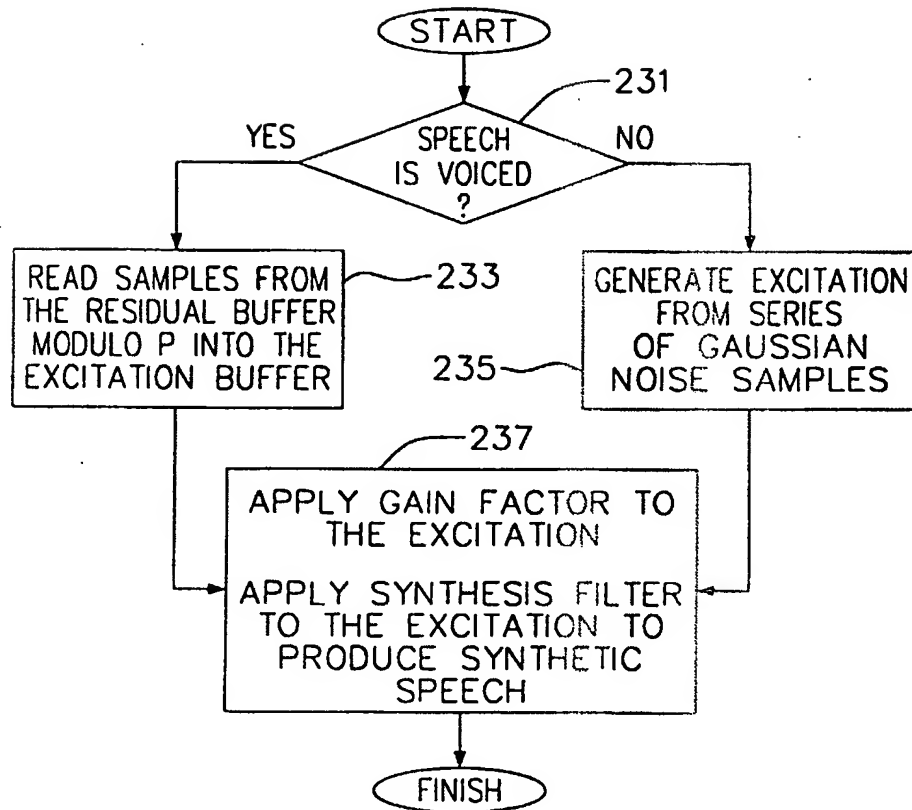


FIG. 14

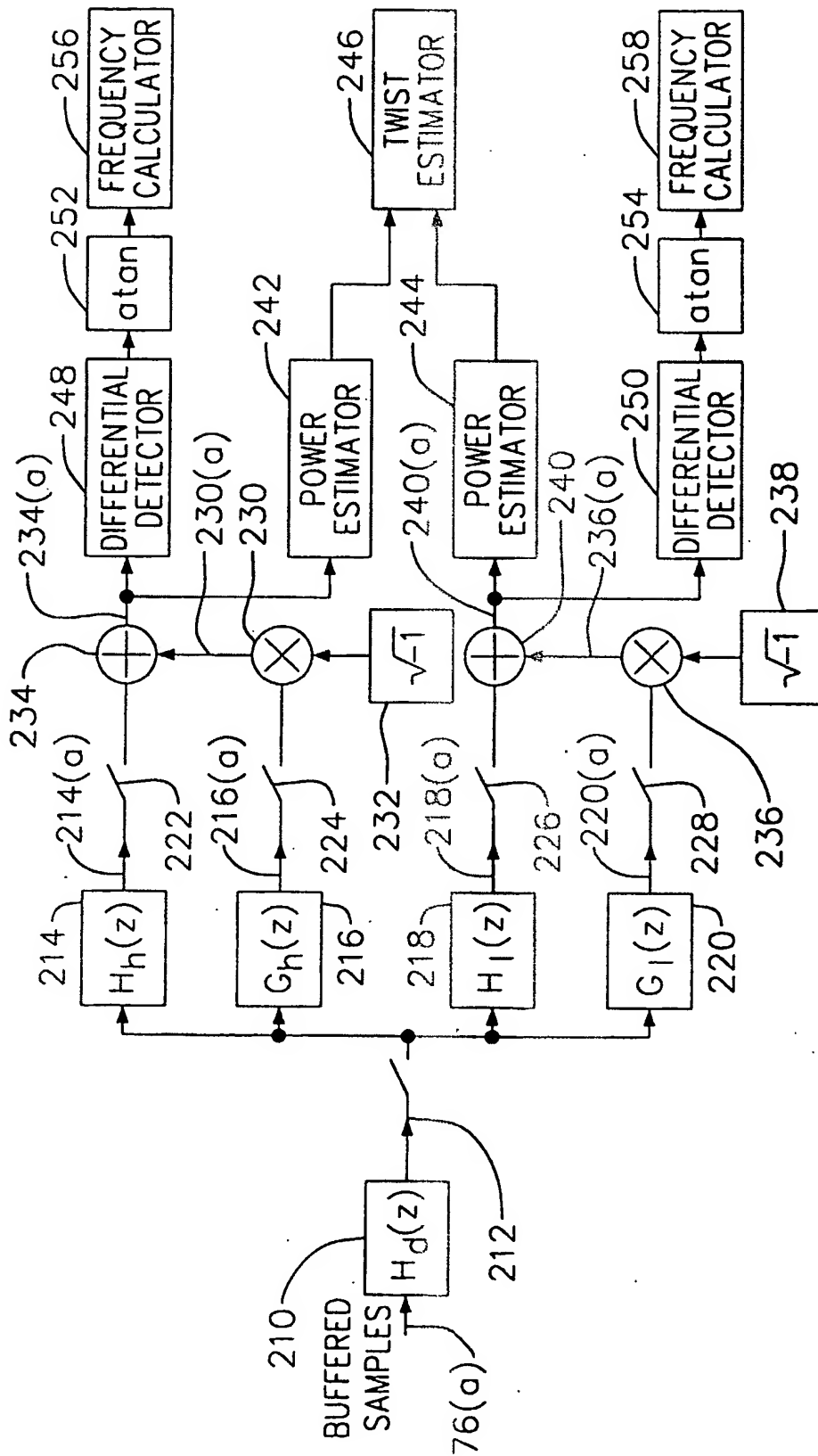


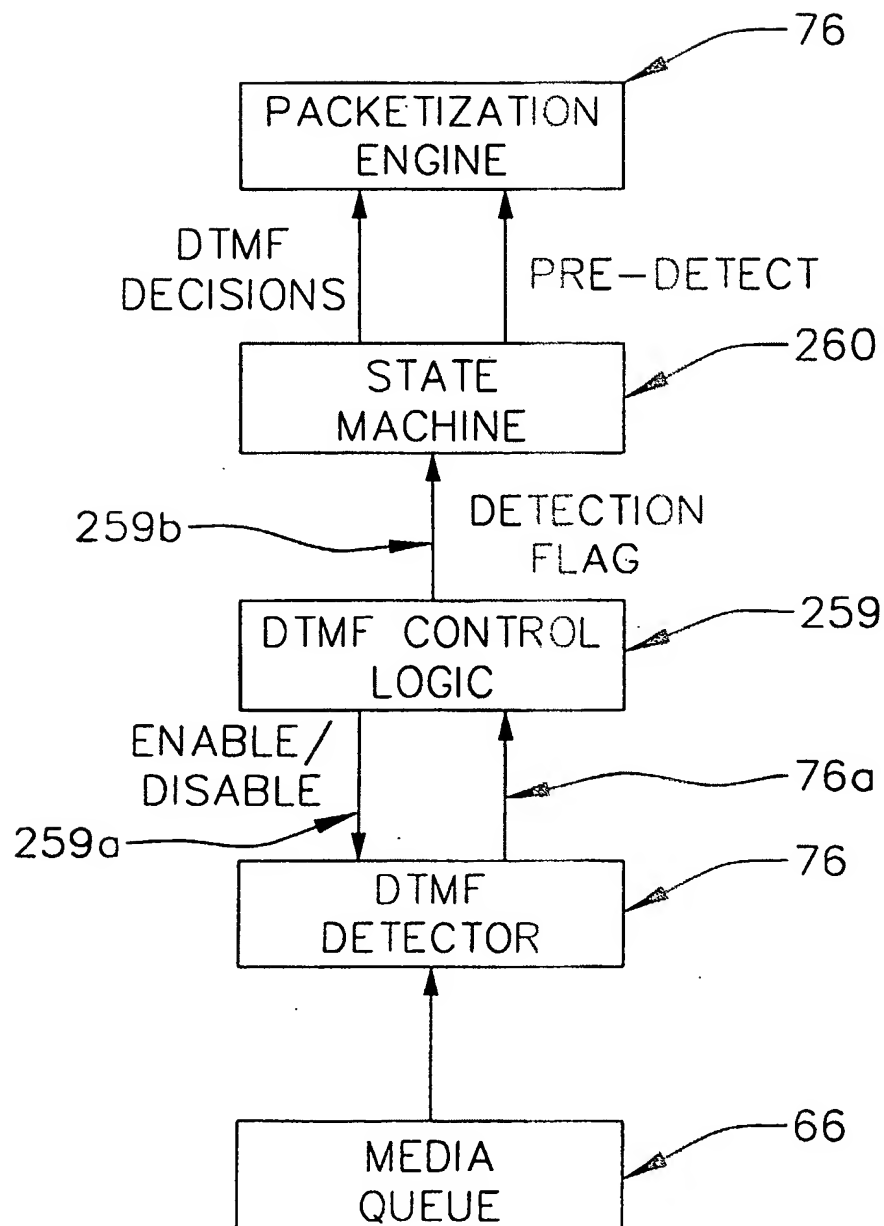
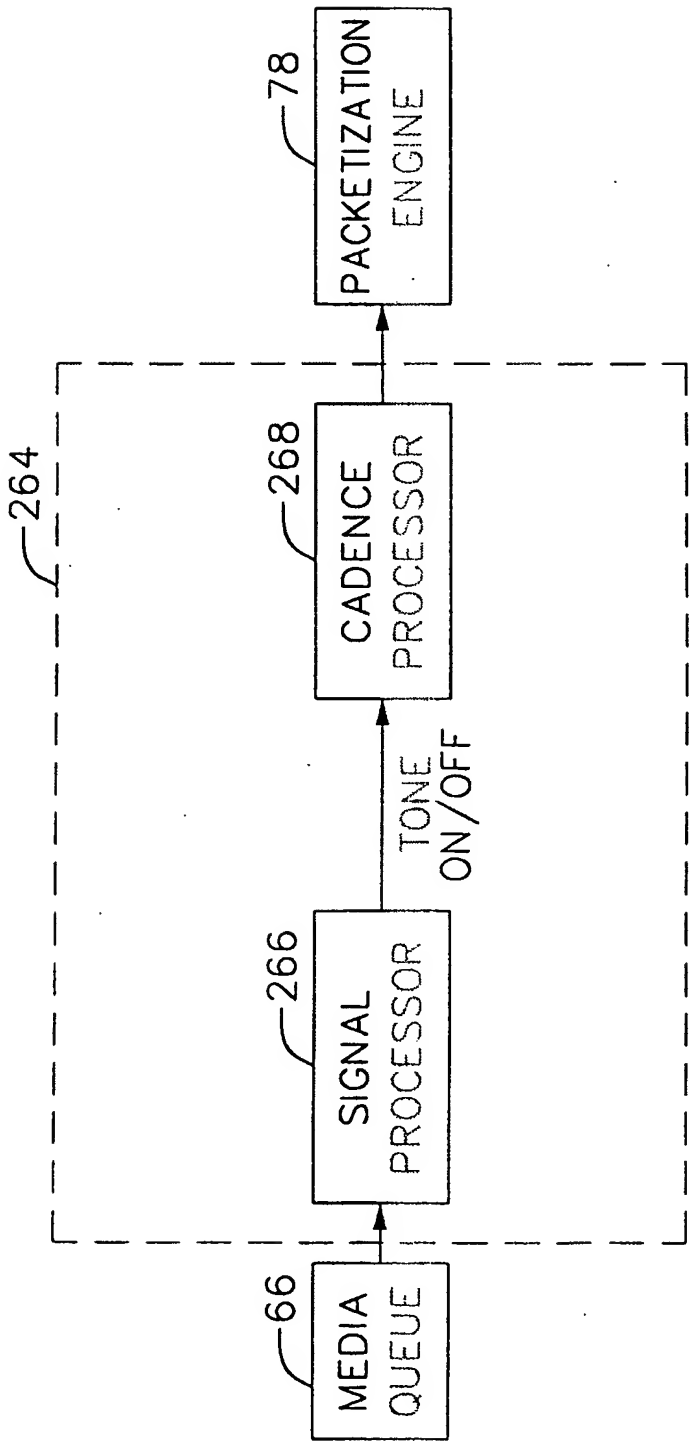
FIG. 14A

FIG. 15



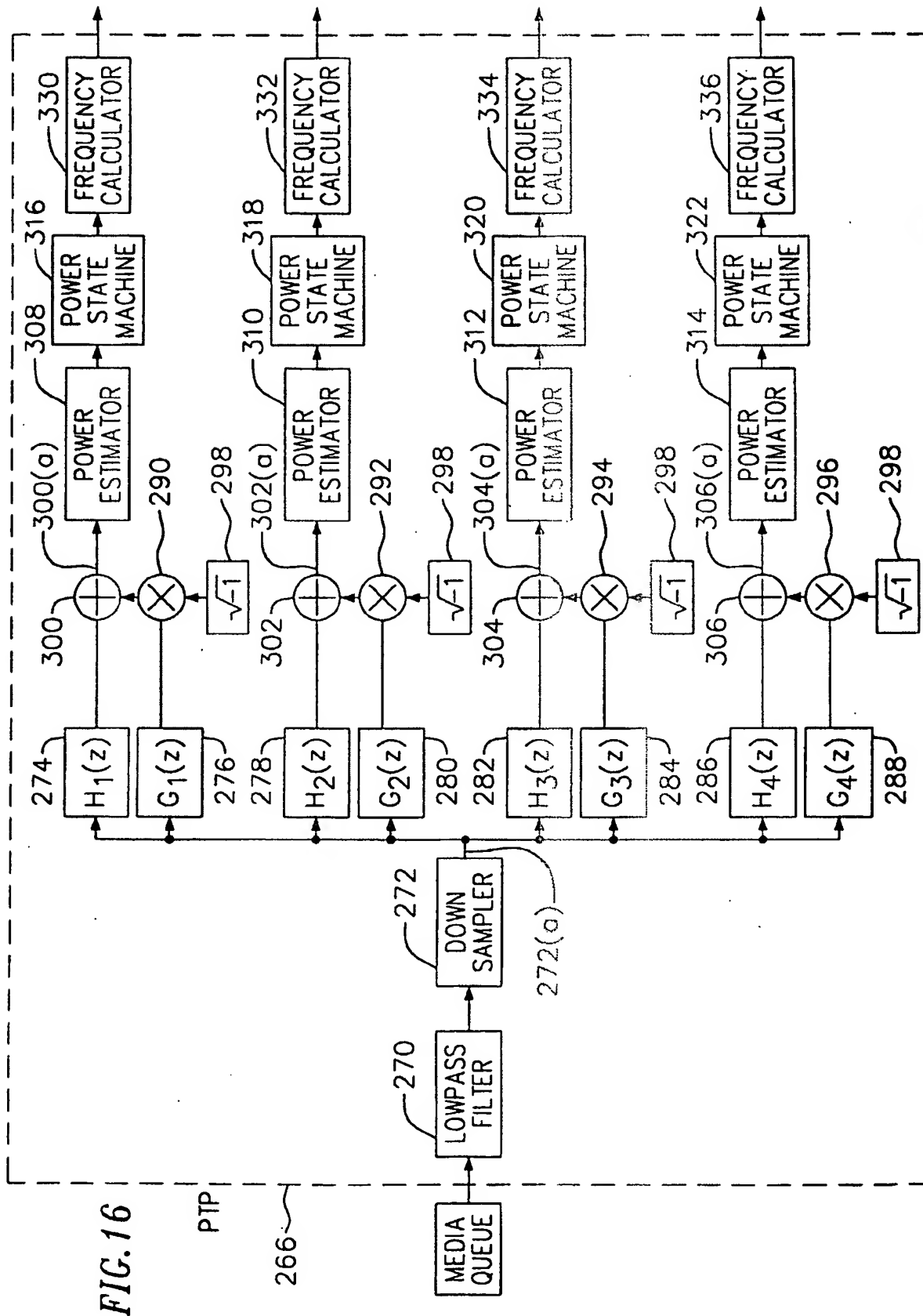


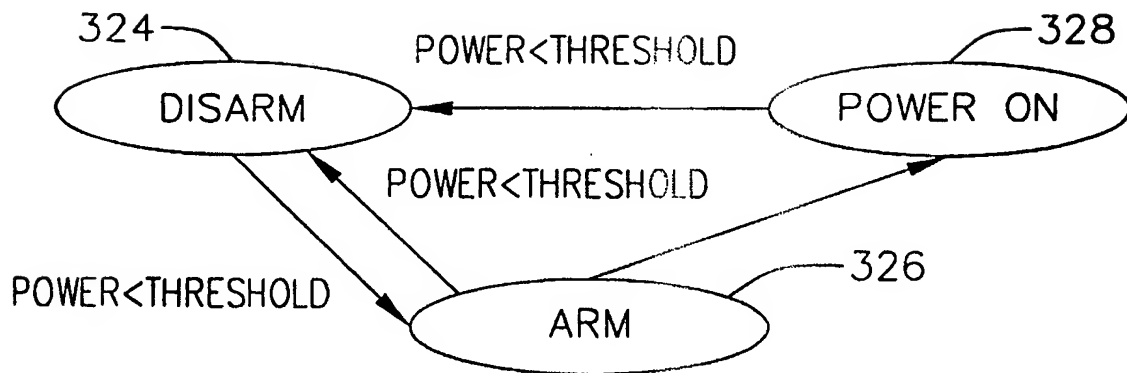
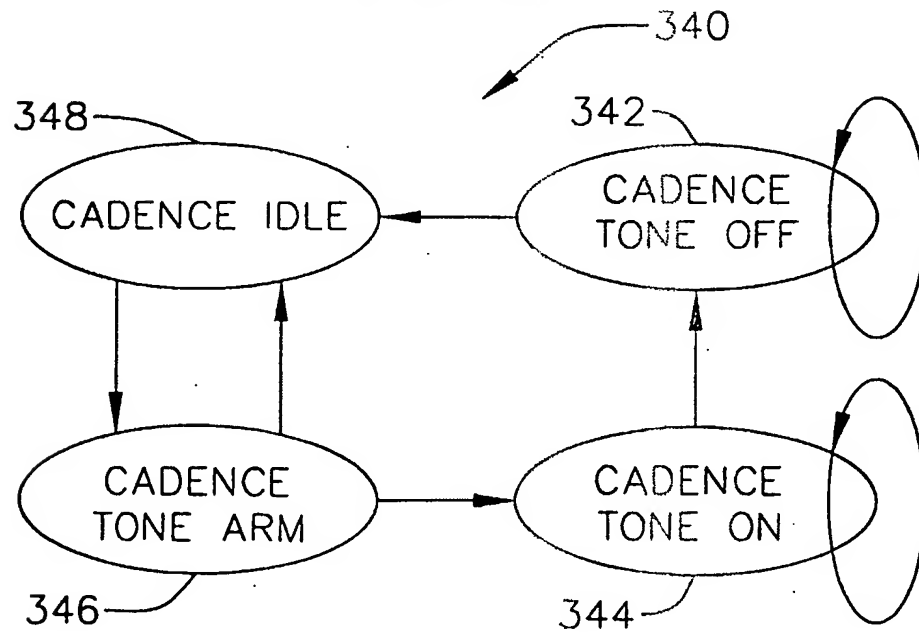
FIG. 17*FIG. 18*

FIG. 18A

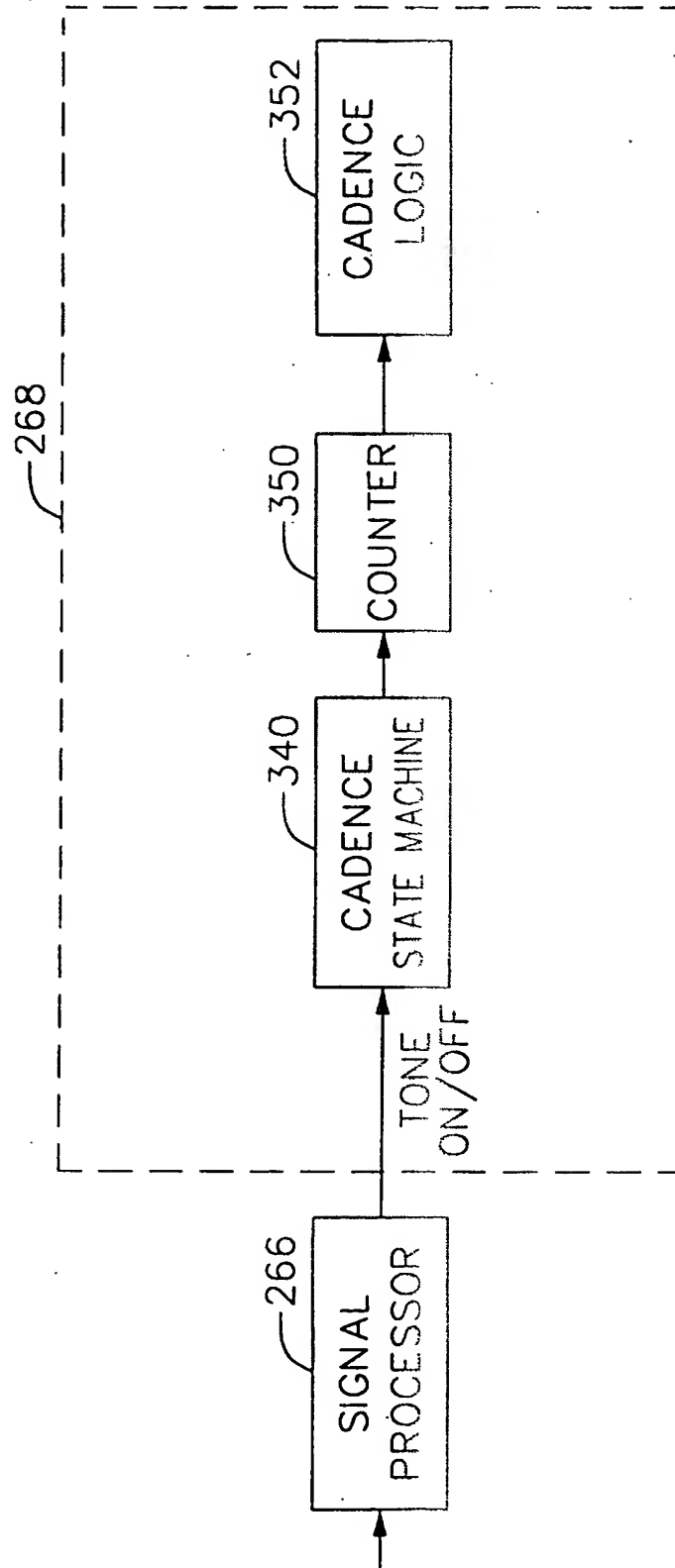


FIG. 19

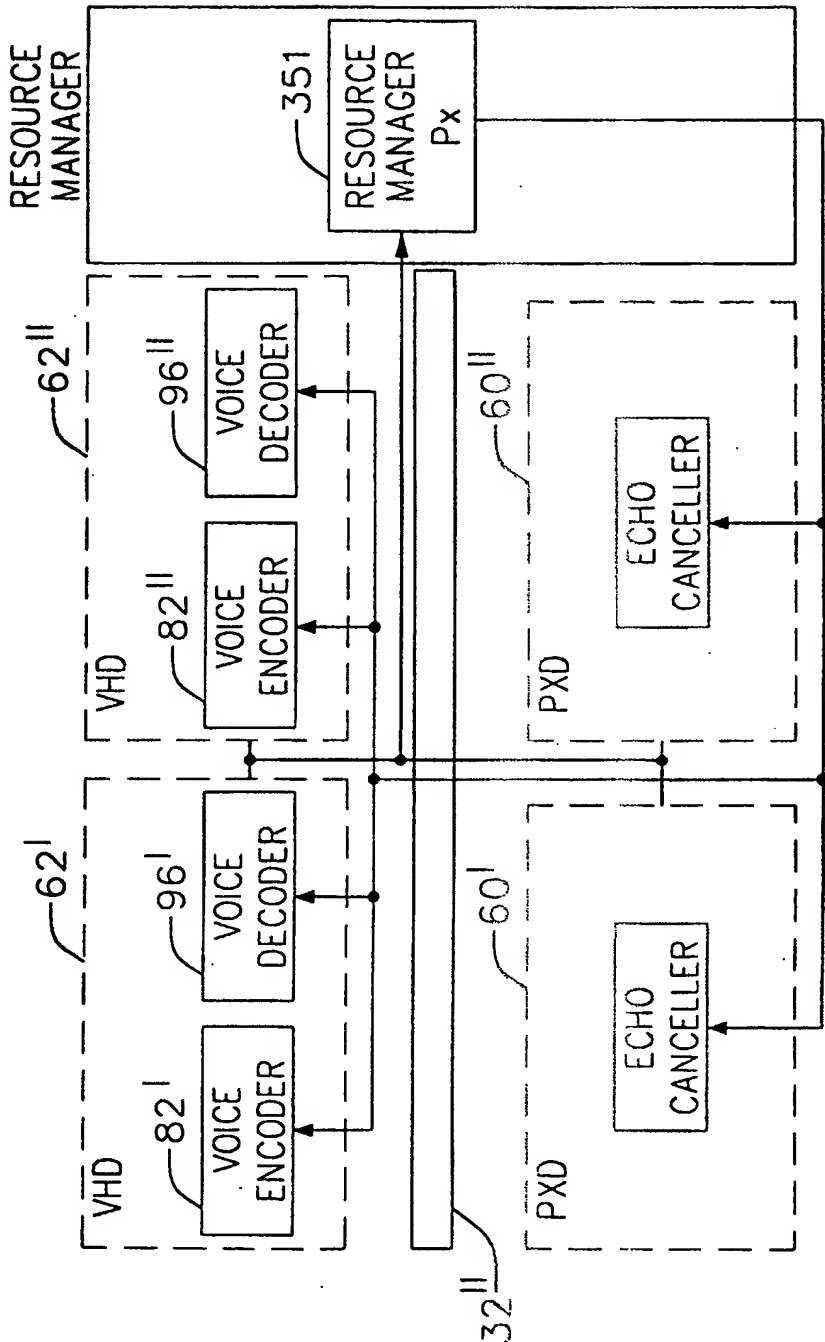


FIG. 20

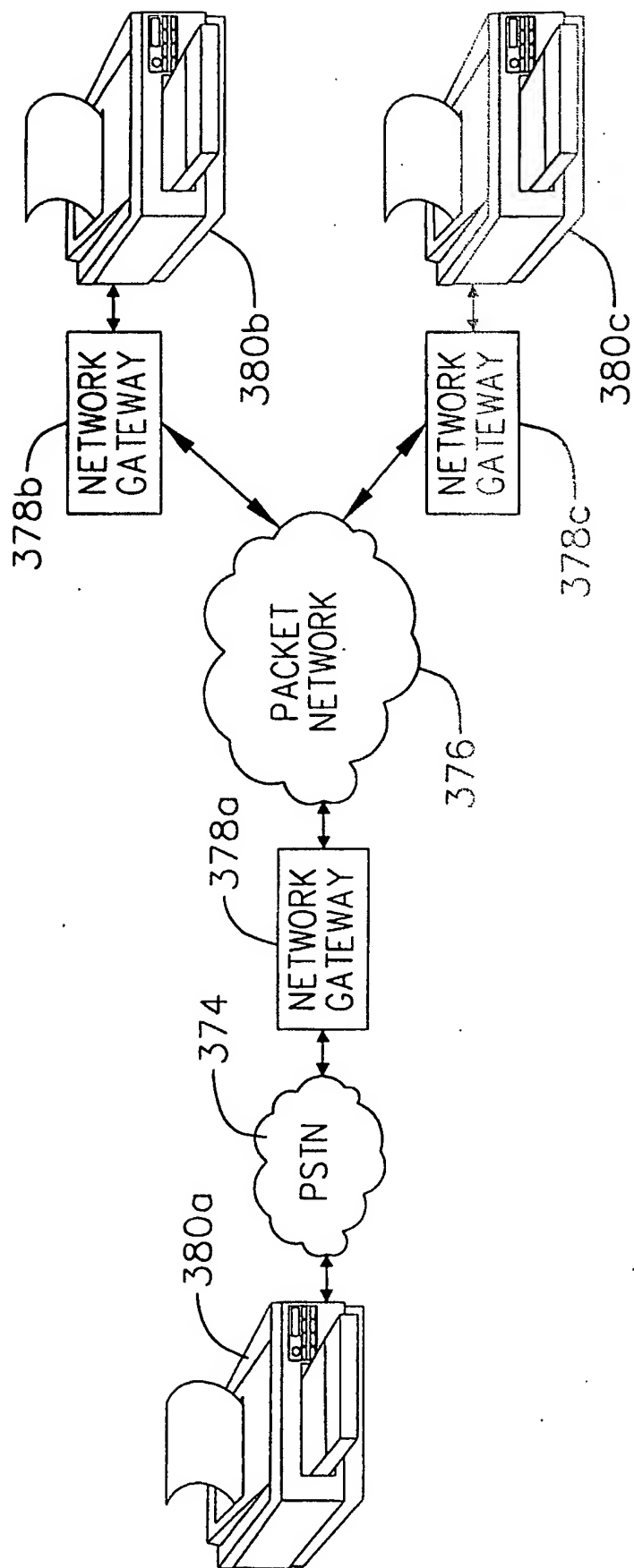


FIG. 21

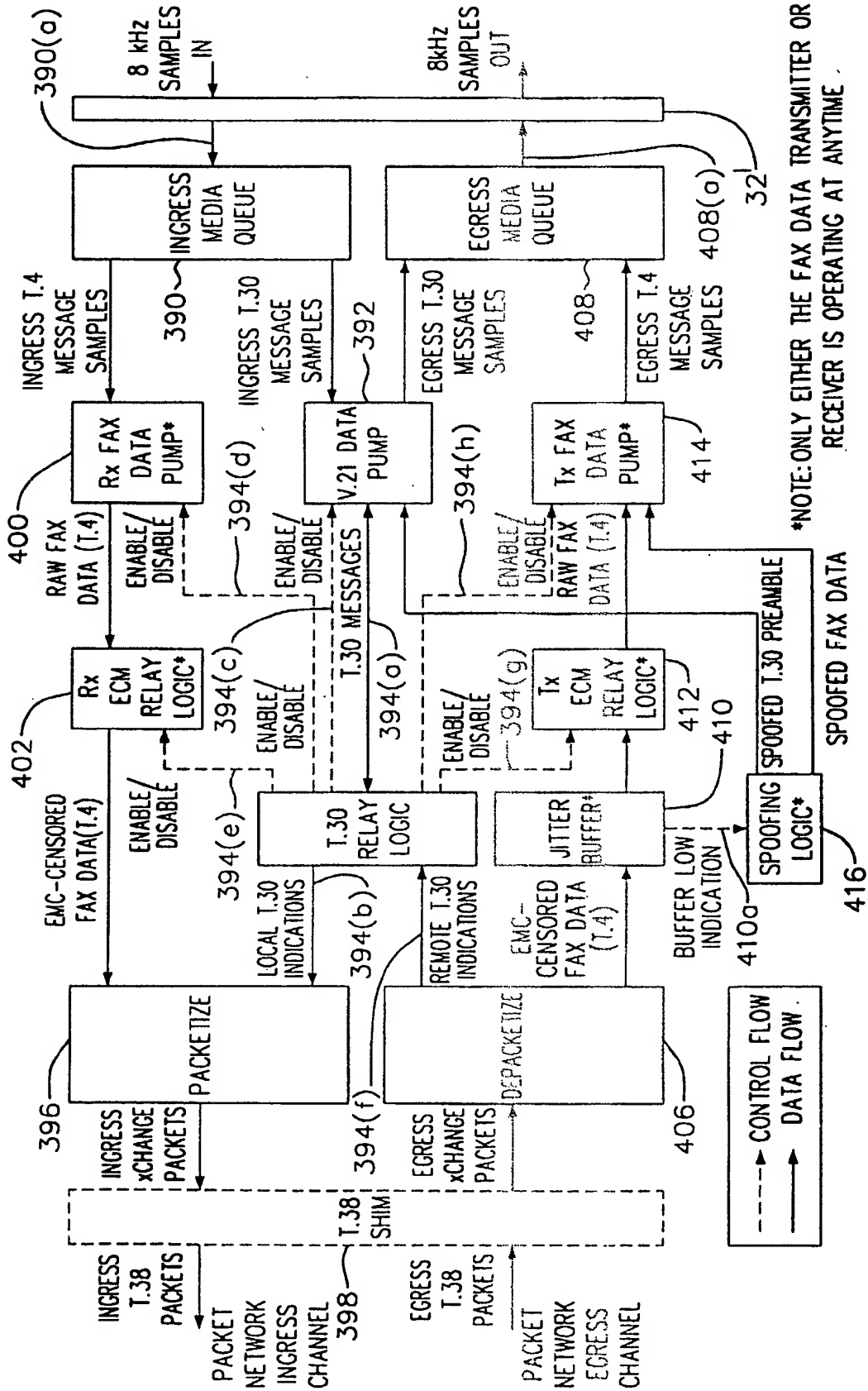


FIG.22

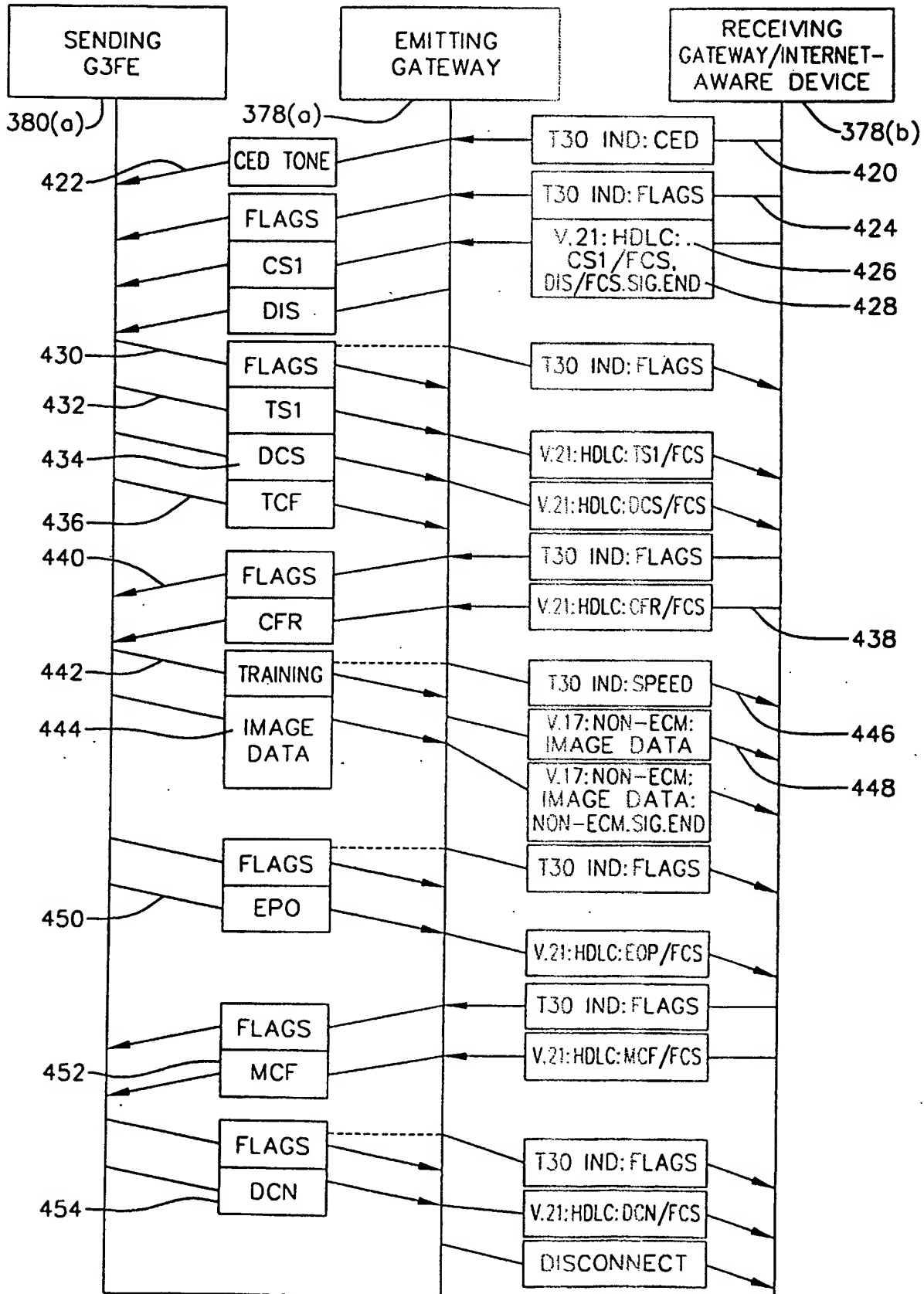


FIG. 23

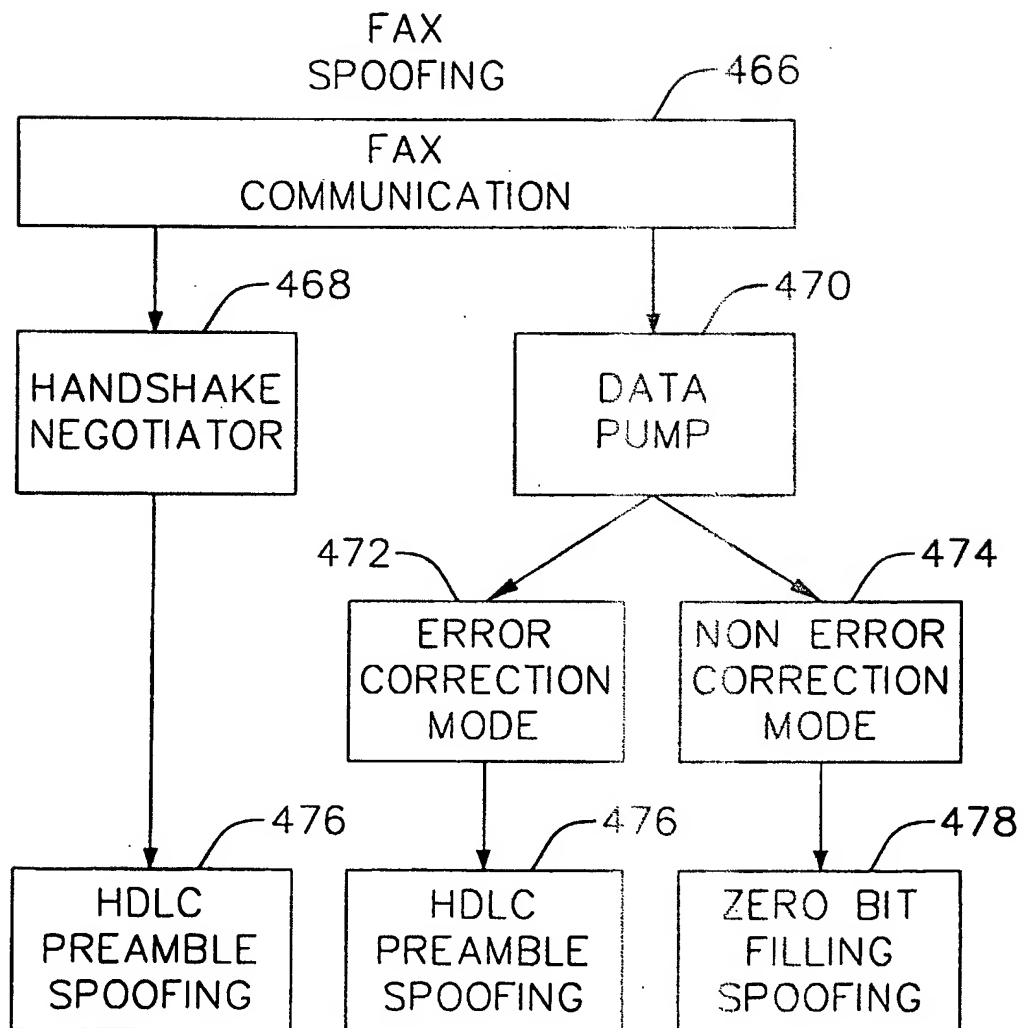


FIG. 24

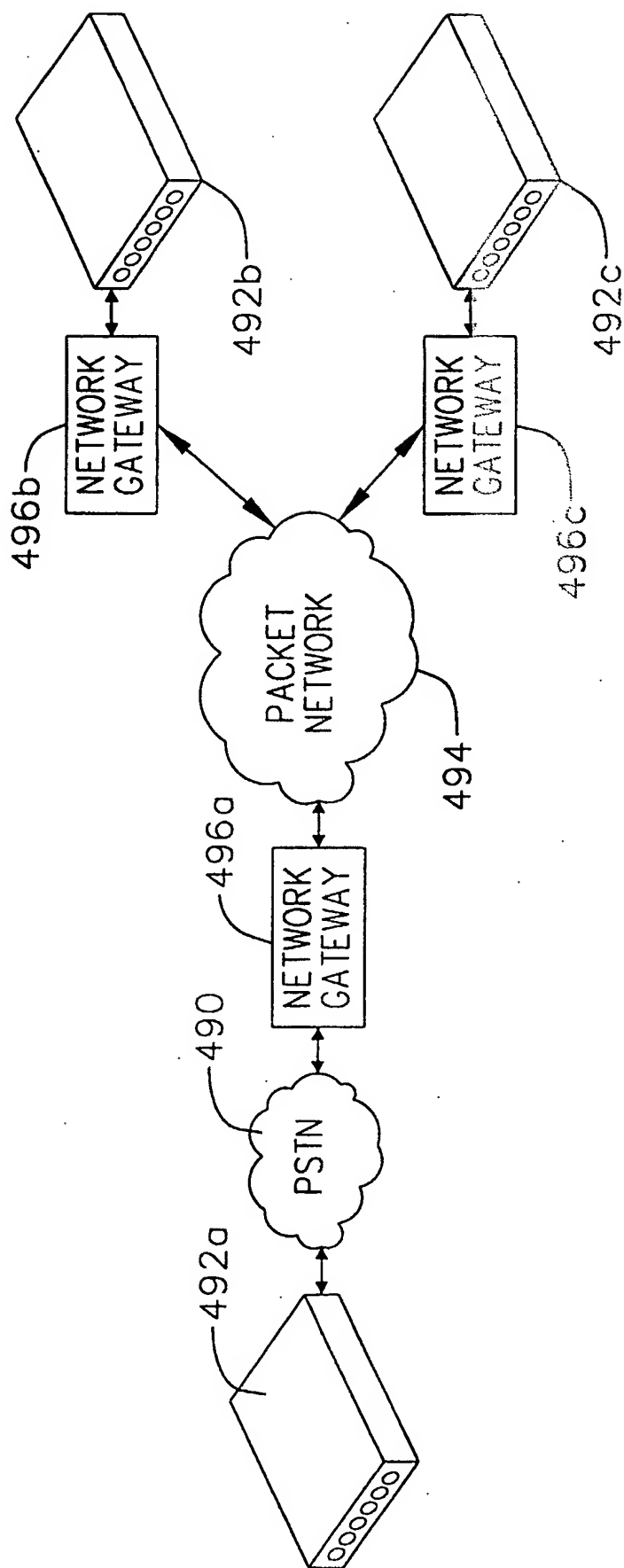


FIG. 25

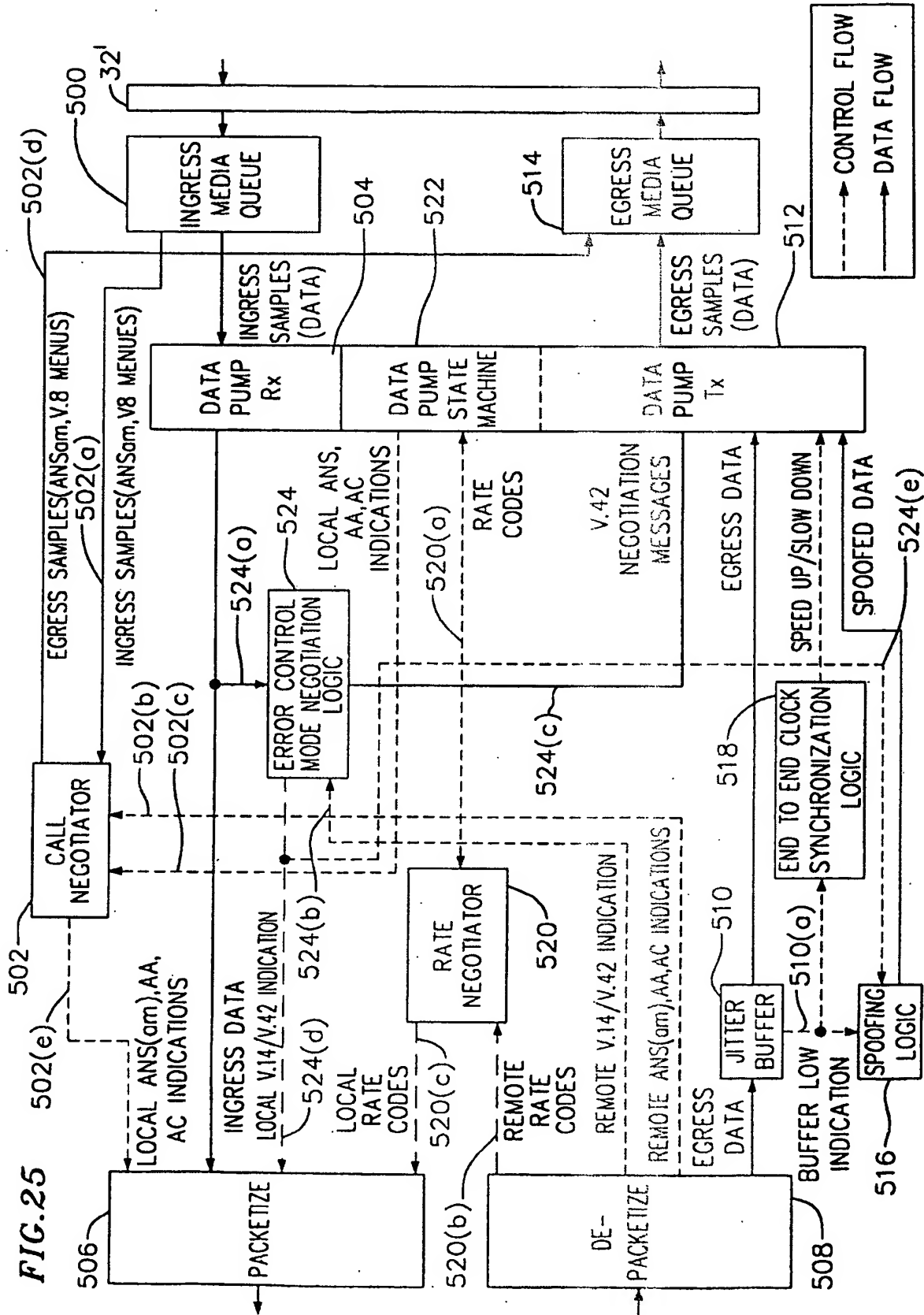


FIG. 26

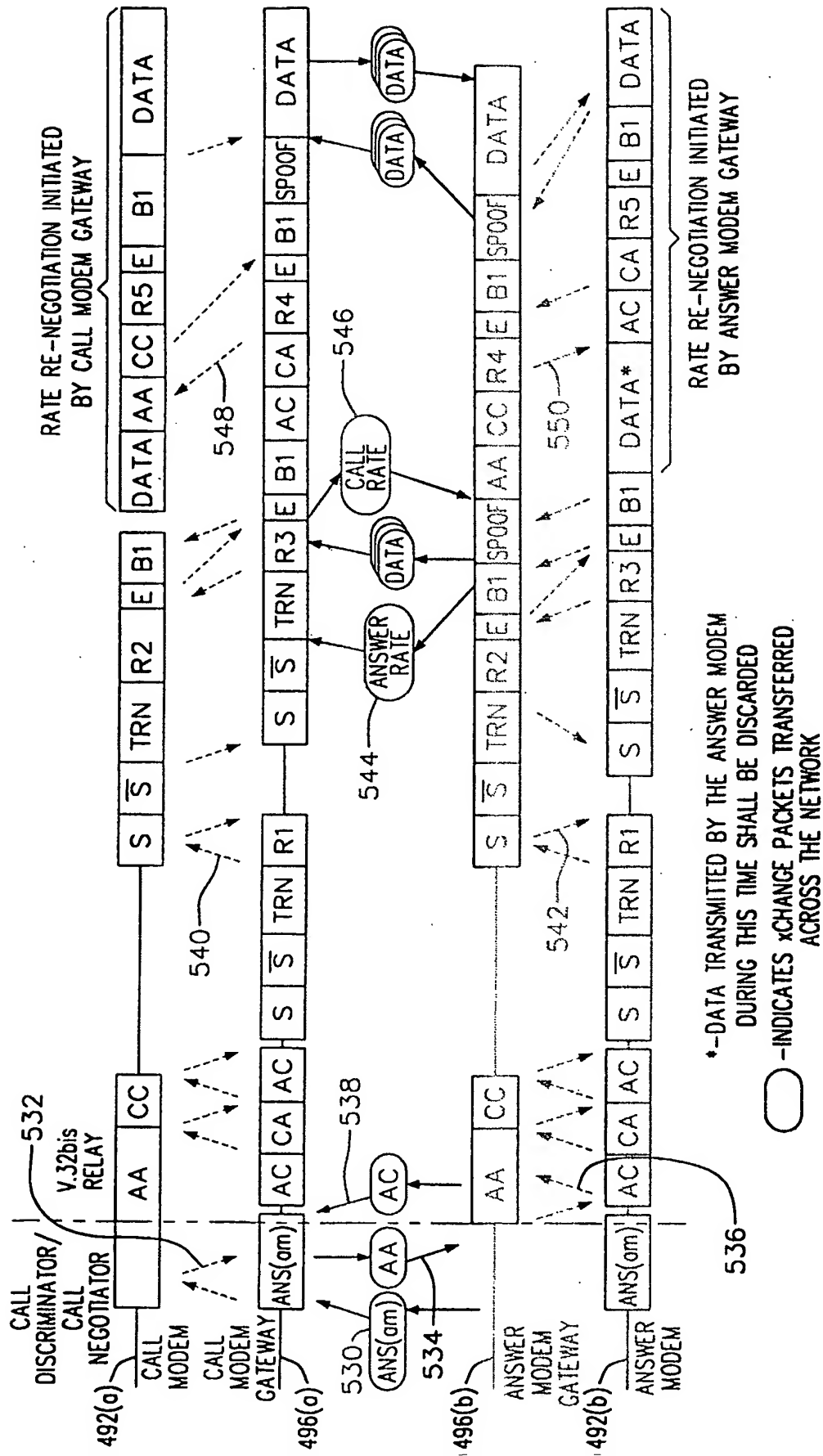


FIG. 27

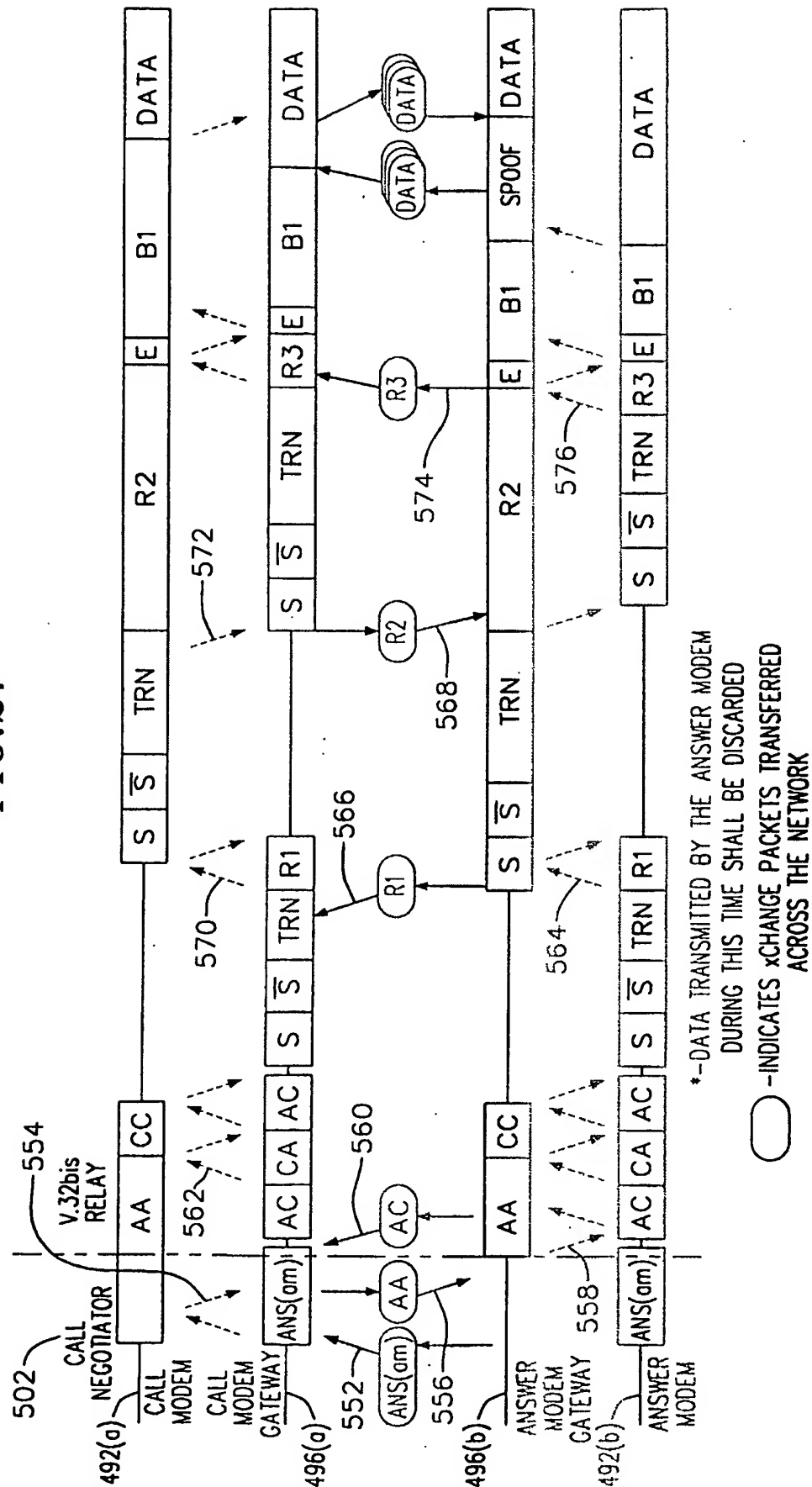


FIG. 28

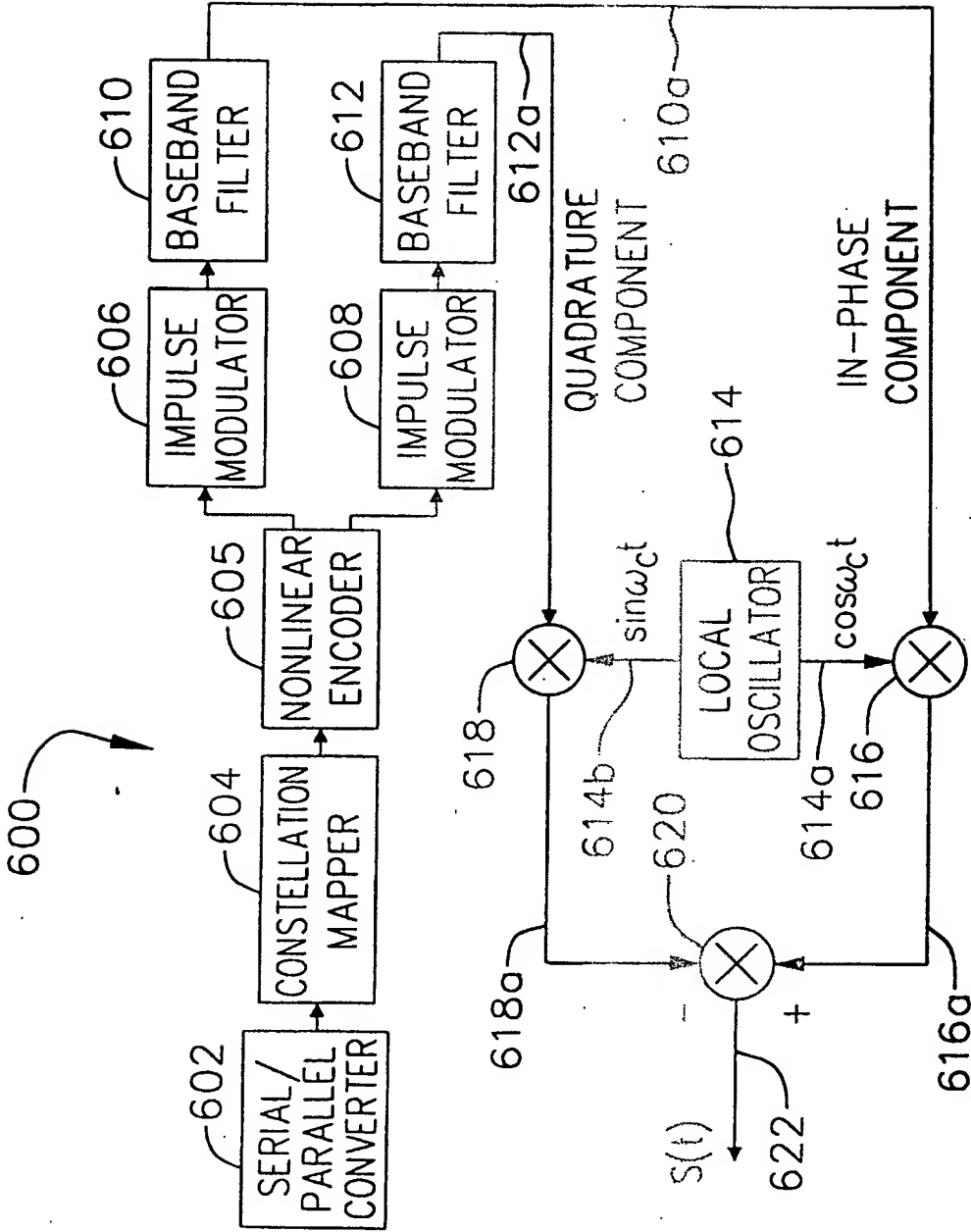
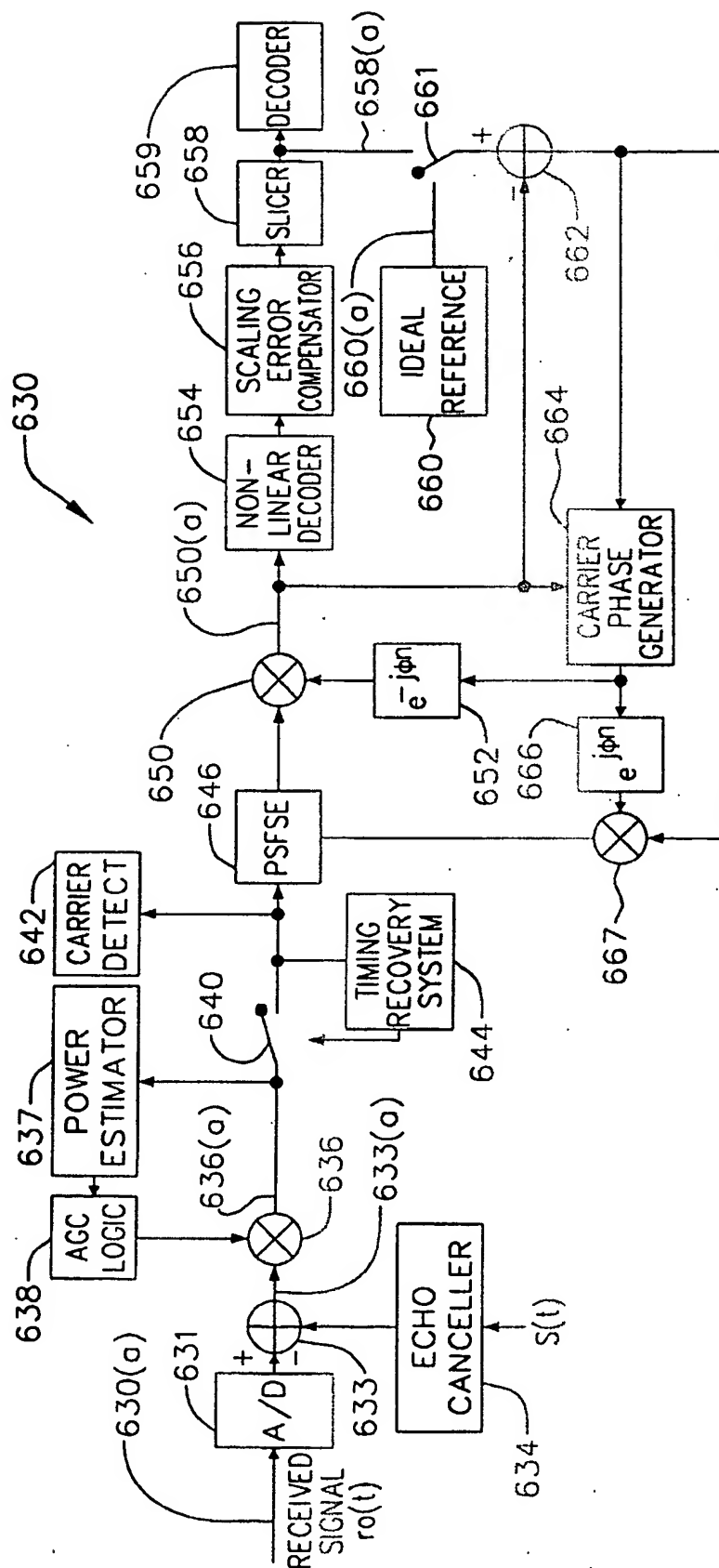


FIG. 29



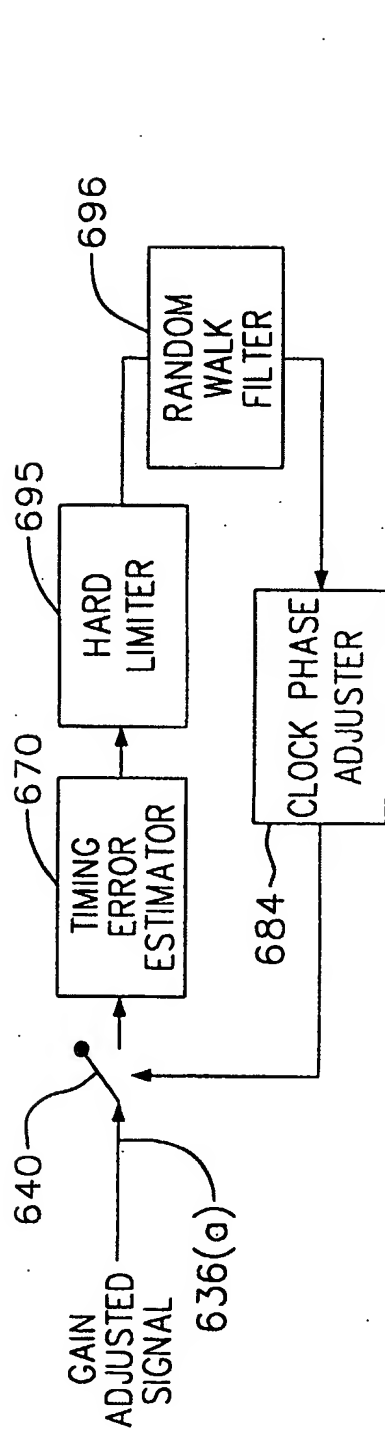


FIG. 32

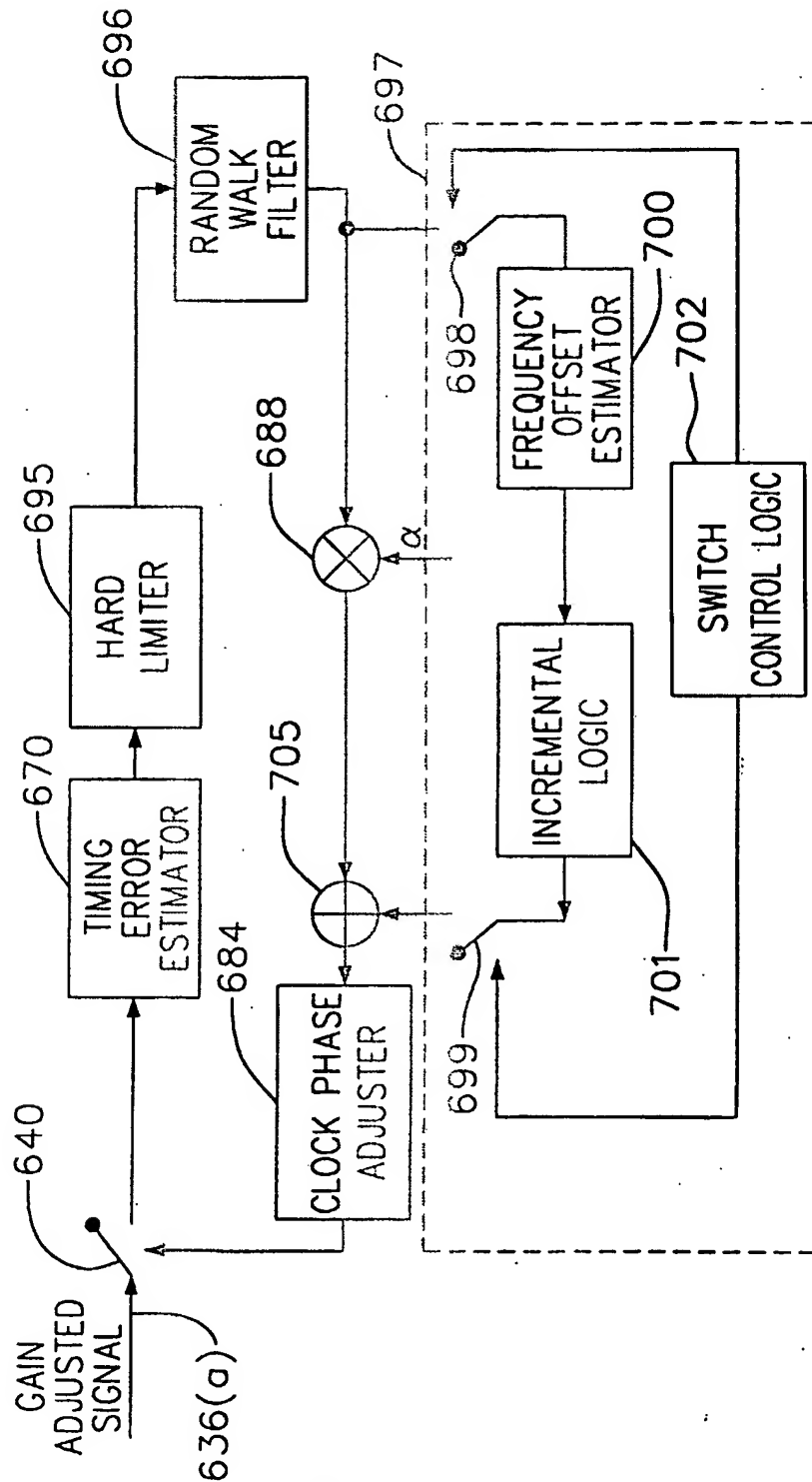


FIG. 33

FIG. 34

700

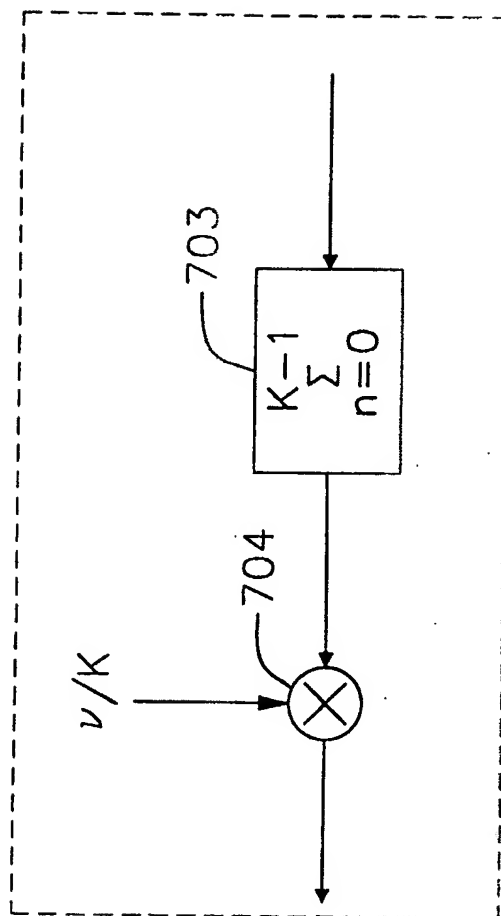
TIMING FREQUENCY
OFFSET ESTIMATE

FIG. 35

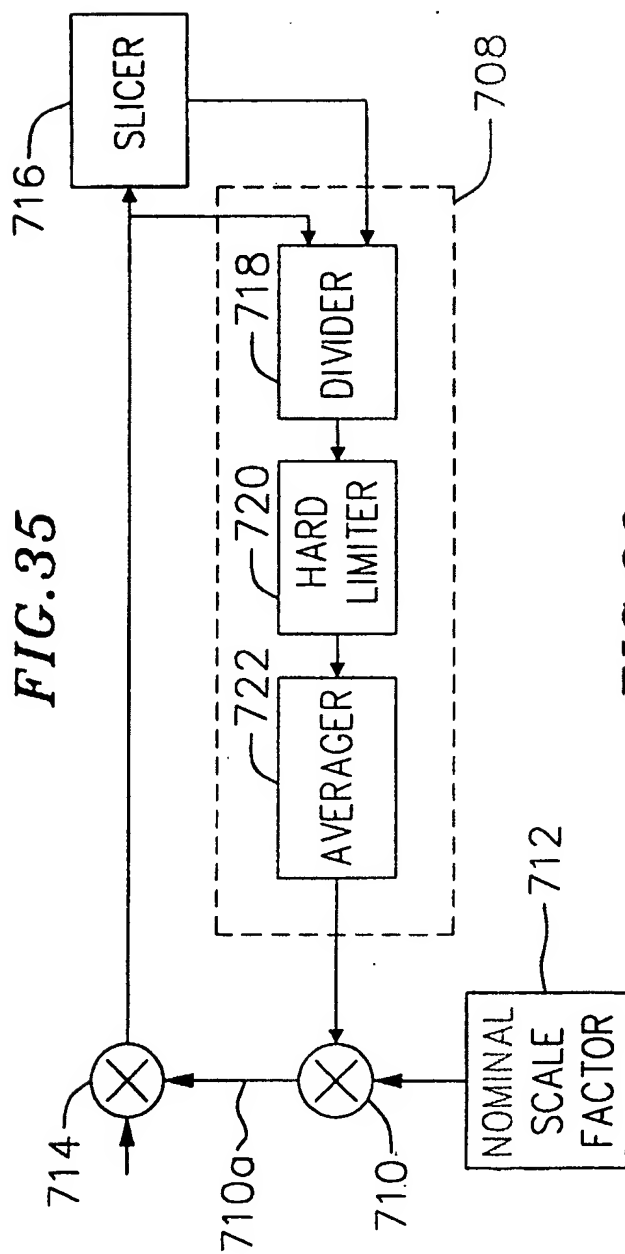
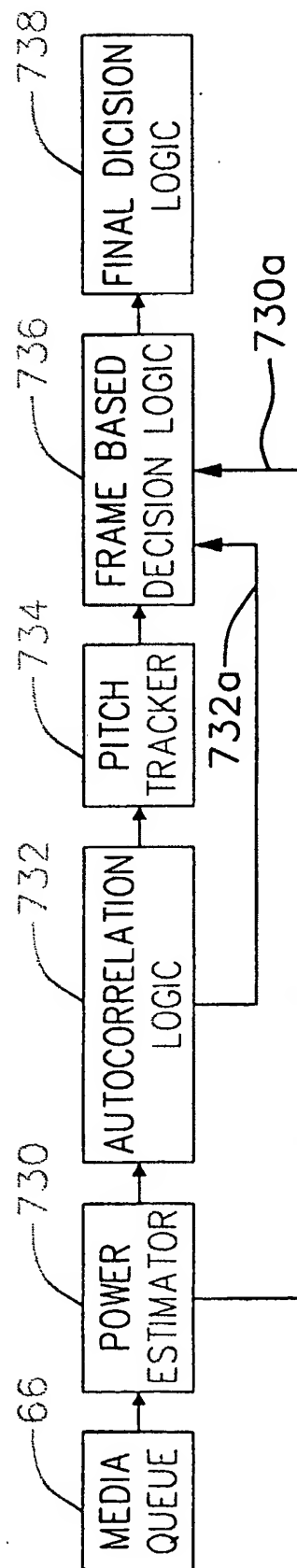


FIG. 36



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.